

TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA TEXTILNÍ

DIPLOMOVÁ PRÁCE

Liberec 2007

ROMAN BUREŠ

TECHNICKÁ UNIVERZITA V LIBERCI
FAKULTA TEXTILNÍ

Katedra oděvnictví

Obor: Řízení technologických procesů

AUTOMATICKÉ ROZPOZNÁVÁNÍ A KLASIFIKACE
V OBRAZOVÝCH DATABÁZÍCH

Automatic recognition and classification in image database

Roman Bureš

VEDOUCÍ DIPLOMOVÉ PRÁCE: Doc. RNDr. Aleš Linka, Csc.

ŠKOLNÍ ROK: 2006/2007

ROZSAH PRÁCE A PŘÍLOH:	74
POČET STRAN:	69
POČET OBRÁZKŮ:	29
POČET TABULEK:	13
POČET PŘÍLOH:	4

P r o h l á š e n í

Prohlašuji, že předložená diplomová práce je původní a zpracoval/a jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem v práci neporušil/a autorská práva (ve smyslu zákona č. 121/2000 Sb. O právu autorském a o právech souvisejících s právem autorským).

Souhlasím s umístěním diplomové práce v Univerzitní knihovně TUL.

Byl/a jsem seznámen/a s tím, že na mou diplomovou práci se plně vztahuje zákon č.121/2000 Sb. o právu autorském, zejména § 60 (školní dílo).

Beru na vědomí, že TUL má právo na uzavření licenční smlouvy o užití mé diplomové práce a prohlašuji, že **s o u h l a s í m** s případným užitím mé diplomové práce (prodej, zapůjčení apod.).

Jsem si vědom toho, že užít své diplomové práce či poskytnout licenci k jejímu využití mohu jen se souhlasem TUL, která má právo ode mne požadovat přiměřený příspěvek na úhradu nákladů, vynaložených univerzitou na vytvoření díla (až do jejich skutečné výše).

V Liberci, dne 14.5.2007

.....

Roman Bureš

Poděkování

Touto cestou bych rád poděkoval vedoucímu diplomové práce Doc. RNDr. Aleši Linkovi, Csc. a konzultantovi Ing. Maroši Tunákovi za odborné usměrnění a metodickou pomoc při realizaci této diplomové práce diplomové.

AUTOMATICKÉ ROZPOZNÁVÁNÍ A KLASIFIKACE V OBRAZOVÝCH DATABÁZÍCH

Abstrakt:

Tato diplomová práce se zabývá automatickým vyhledáváním v referenční databázi otisků podešví bot. Vyhledávání je zaměřeno na využití metody LSI (Latent Semantics Indexing), dvojrozměrné Fourierově transformaci a centrálních momentů. Metoda LSI využívá singulárního rozkladu matice dokumentů, která je složena z vektorů obrázků (extrakcí vlastností obrázků). Rozpoznávání vyhledávaného obrázku je realizováno pomocí kosinové míry podobnosti. Postup vyhledávání využívající dvojrozměrnou Fourierovu transformaci převede rozložení obrazových intenzit $f(x,y)$ vyhodnocovaného obrazu na obraz prostorových frekvencí $F(f_x, f_y)$, tzv. obrazové spektrum. Vyhledávání pak spočívá v korelaci mezi spektrem vyhledávaného obrázku a spektry jednotlivých obrázků databáze. Metoda centrálních momentů produkuje soubor hlavních vektorů, které jsou invariantní vůči posunutí, natočení a změně velikosti obrazu. Ke klasifikaci pro metodu centrálních momentů je použito klasifikačního stromu.

AUTOMATIC RECOGNITION AND CLASSIFICATION IN IMAGE DATABASE

Abstract:

In the thesis automatic recognition and classification in image database of shoeprints is being solved. The searching is oriented on the application of the LSI method (Latent Semantic Indexing), on Fourier transformation and on the using central moments. LSI method is using for singular analysis of matrix documents. This one is composed from the vector images (extraction of image properties). A recognition of the found image is being carried out using cosine similarity measure. The finding procedure employing two dimensional Fourier transform is transforming the distribution of image intensities of $f(x,y)$ of evaluated image on the space frequencies $F(f_x, f_y)$, called image spectra. The part of recognition is also correlated between the spectra found image and spectra of the single database images. Central moment method is producing a set of fundamental vectors, which are invariant to the displacement, to rotation and to the image extend. To the classification of the central moments the classification tree has been used.

Klíčová slova:

Latentní sémantické indexování
Diskrétní Fourierova transformace
Invariantní momenty
Obrazová databáze
Otisky podešví bot
Rozpoznávání
Klasifikace

Key words:

Latent Semantic Indexing
Discrete Fourier transform
Invariant moments
Image database
Shoeprints
Recognition
Classification

1	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	10
2	ÚVOD	12
2.1	EXTRAKCE CHARAKTERISTICKÝCH RYSŮ	13
2.1.1	<i>Barva</i>	<i>13</i>
2.1.2	<i>Struktura - vazba</i>	<i>13</i>
2.1.3	<i>Tvar</i>	<i>14</i>
2.2	VYSOKODIMENSIONÁLNÍ INDEXOVÁNÍ	14
2.3	SYSTÉMY PRO OBRAZOVÉ VYHLEDÁVÁNÍ	14
2.4	SMĚRY VÝZKUMU V BUDOUCNU	15
3	POPIS PROBLÉMU	16
4	SHRNUTÍ SOUČASNÉHO STAVU.....	18
5	DIGITÁLNÍ OBRAZ.....	19
5.1	OBRAZOVÁ STRUKTURA	19
5.2	PRAHOVÁNÍ OBRAZU	20
5.3	KORELACE OBRAZŮ	21
5.4	NATOČENÍ OBRAZU	21
5.5	GAUSSOVSKÝ, BÍLÝ ŠUM.....	22
6	ROZDĚLENÍ DATABÁZE DO 9-TI PODSKUPIN	24
6.1	PŘEVZORKOVÁNÍ OBRAZŮ DATABÁZE NA STEJNOU VELIKOST :	24
6.2	NALEZENÍ ORIGIN POINTU (DĚLÍČHO BODU).....	24
7	TEORETICKÁ ČÁST	28
7.1	LATENT SEMANTIC INDEXING (LSI)	28
7.1.1	<i>Extrakce vektorů vlastností pomocí metody LSI.....</i>	<i>28</i>
7.1.2	<i>Singulární rozklad matice dokumentů A.....</i>	<i>29</i>
7.1.3	<i>Kosinová míra podobnosti</i>	<i>30</i>
7.1.4	<i>Vyhledávání vektoru v matici dokumentů.....</i>	<i>31</i>
7.1.5	<i>Použití LSI pro databázi obrázků.....</i>	<i>32</i>
7.1.6	<i>Rekonstrukce obrázků</i>	<i>32</i>
7.2	DISKRÉTNÍ FOURIEROVA TRANSFORMACE.....	35
7.2.1	<i>2D diskrétní Fourierova transformace.....</i>	<i>36</i>
7.2.2	<i>Posunutí</i>	<i>37</i>
7.2.3	<i>Spektrální přístup k vyhledávání v referenční databázi</i>	<i>38</i>
7.3	METODA INVARIANTNÍCH MOMENTŮ	39
7.3.1	<i>Centrální momenty.....</i>	<i>40</i>
7.3.2	<i>Klasifikační stromy.....</i>	<i>43</i>
8	EXPERIMENTÁLNÍ ČÁST	44
8.1	EXPERIMENT 1 - POUŽITÍ METODY LSI NA VYBRANOU SKUPINU DESETI STOP	44
8.1.1	<i>Závěr experimentu 1</i>	<i>46</i>
8.2	EXPERIMENT 2 - POUŽITÍ METODY LSI NA VYBRANOU SKUPINU DESETI POŠKOZENÝCH STOP.....	47
8.2.1	<i>Závěr experimentu 2.....</i>	<i>49</i>
8.3	EXPERIMENT 3 - POUŽITÍ METODY LSI NA REÁLNOU STOPU	49
8.3.1	<i>Závěr experimentu 3.....</i>	<i>54</i>
8.4	EXPERIMENT 4 - POUŽITÍ METODY DFT NA VYBRANOU SKUPINU DESETI STOP	54
8.4.1	<i>Závěr experimentu 4.....</i>	<i>57</i>
8.5	EXPERIMENT 5 - POUŽITÍ METODY DFT NA VYBRANOU SKUPINU DESETI POŠKOZENÝCH STOP	57
8.5.1	<i>Závěr experimentu 5.....</i>	<i>59</i>
8.6	EXPERIMENT 6 - POUŽITÍ METODY DFT PRO VYHLEDÁVÁNÍ REALNÉ STOPY	59
8.6.1	<i>Závěr experimentu 6.....</i>	<i>62</i>
8.7	EXPERIMENT 7 - POUŽITÍ METODY CENTRÁLNÍCH MOMENTŮ A KLASIFIKAČNÍHO STROMU	62
8.7.1	<i>Závěr experimentu 7.....</i>	<i>64</i>
8.8	EXPERIMENT 8 - POUŽITÍ SVD ROZKLADU A KLASIFIKAČNÍHO STROMU PRO VYHLEDÁVÁNÍ V REFERENČNÍ DATABÁZI.....	65
8.8.1	<i>Závěr experimentu 8.....</i>	<i>65</i>

9	ZÁVĚR.....	66
10	LITERATURA	68
11	PŘÍLOHY	70
11.1	PŘÍLOHA 1: <i>LSI.M</i> – M-FILE PRO METODU LSI.....	71
11.2	PŘÍLOHA 2: <i>DFT.M</i> – M-FILE PRO METODU DFT	72
11.3	PŘÍLOHA 3: <i>CMTREE.M</i> – M-FILE PRO KLASIFIKAČNÍ STROM Z MATICE CENTRÁLNÍCH MOMENTŮ DATABÁZE.....	74
11.4	PŘÍLOHA 4: <i>LSITREE.M</i> – M-FILE PRO KLASIFIKAČNÍ STROM Z MATICE $V_{K=48}$;	74

1 Seznam použitých symbolů a zkratek

RGB	- Red-Green-Blue
LSI	- Latent Semantic Indexing
DFT	- Discrete Fourier Transfor
FT	- Fourier Transform
$1D$	- Jednorozměrné
$2D$	- Dvojměrné
FFT	- Fast Fourier Transform = rychlá Fourierova transformace
$f(x,y)$	- Obrazová funkce
$F(f_x, f_y)$	- Prostorová frekvence
$GB\check{S}$	- Gaussovský Bílý Šum
m	- Počet řádků matice (šířka obrazu)
n	- Počet sloupců matice (délka obrazu)
A	- Obrazová matice A
B	- Obrazová matice B
r	- Korelační koeficient
\bar{A}	- Aritmetický průměr hodnot matice A
\bar{B}	- Aritmetický průměr hodnot matice B
$S(\omega)$	- Spektrální hustota
OP	- Origin point
s_1	- První krajní sloupec, kde zasahuje objekt
s_2	- Poslední krajní sloupec, kde zasahuje objekt
r_1	- První krajní řádek, kde zasahuje objekt
r_2	- Poslední krajní řádek, kde zasahuje objekt
PS	- Počet stop v databázi
(r,s)	- Souřadnice Origin Pointu
U	- Matice vektorů konceptů
S	- Diagonální matice singulárních čísel
V^T	- Matice pseudo-dokumentů
k	- Velikost báze prostoru po SVD rozkladu
U_k	- Matice vektorů konceptů dimenze k po SVD rozkladu
S_k	- Diagonální matice singulárních čísel dimenze k po SVD rozkladu

V_k	- Matice pseudo-dokumentů dimenze k po SVD rozkladu
q	- projekce vyhledávaného vektoru jeho do U_k
qc	- Vektor dotazu (vektor vyhledávaného obrazu)
D	- Matice dokumentů
D_i	- i -tý dokument matice D
w_{ij}	- Váha důležitosti
Q	- Vektor dotazu
$Sim(Q, D_i)$	- Kosinová míra podobnosti
$norm(X)$	- Euklidovská vzdálenost vektoru X
$F(u, v)$	- Dvojměrná Fourierova transformace
$f(m, n)$	- Inverzní Fourierova transformace
$R(u, v)$	- Reálná složka dvojměrné Fourierova transformace
$I(u, v)$	- Imaginární složka dvojměrné Fourierova transformace
$P(u, v)$	- Výkonové spektrum
CM	- Centrální moment
CPU	- Počítačový procesor
RAM	- Operační paměť počítače
HDD	- Pevný disk počítače
ms	- Milisekunda
GB	- GigaByte
MB	- MegaByte
PC	- Personal Computer

2 Úvod

V současné době, s prudkým rozvojem informačních technologií, je každý z nás doslova bombardován nepřehledným množstvím multimediálních dat. Mezi tyto data patří textové, obrazové a zvukové dokumenty.

Získávání dat z databází se stává námětem mnoha prací a výzkumných projektů. Do nedávné doby bylo však vždy spojeno vyhledávání dat s alfanumerickými (textovými) databázemi. Rozvoj internetu a firemních sítí vede k stále masovějšímu použití grafických a zvukových databází a tomu odpovídající formy dat. Dostupnost a nové možnosti vedou k nebývalému zájmu a rozšíření použití multimediálních dat. S tím logicky vzniká potřeba jejich vhodného ukládání, vyhledávání a zpracování. Objevují se rozsáhlé databáze obrázků a záznamů zvuků. Ovšem jejich objemnost a rozličná struktura informací nedovoluje vždy použití klasických algoritmů a postupů pro zpracování a vyhledávání. Proto se hledají alternativní cesty jak rychle a efektivně tato data používat.

Proto vyvstává otázka, jakým způsobem tyto multimediální data reprezentovat a organizovat, aby v nich šlo následně vyhledávat a tím získat jednoduše, rychle a efektivně potřebné informace.

Databáze obrázků a videa (obecně nestrukturované databáze) v dnešní době představují objemově nejrychleji rostoucí datové zdroje. Současně s tím narůstá potřeba umět v takovýchto zdrojích efektivně vyhledávat podle obsahu (tj. podle vlastní informace v obrázku). Jelikož struktura ani sémantika obrazových dat není jednoznačná a nelze tedy účinně pokládat dotazy na přesnou shodu, (kde se očekává jako výsledek množina plně významných objektů k dotazu a zbytek je plně bezvýznamných), je potřeba zavést míru podobnosti, která umožní klasifikovat obrázky v databázi k obrázku dotazu jako více či méně významných.

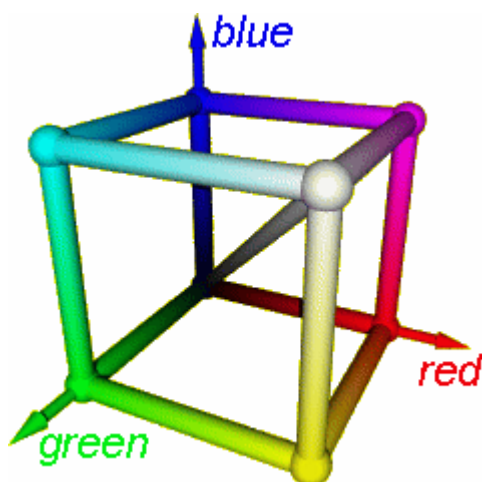
Cílem této práce je nalézt optimální metodu pro vyhledávání v obrazových databázích. Vypracovat postup a algoritmy pro vyhledávání (rozpoznání) objektu v obrazové databázi a to i za předpokladu, že byl obraz dotazu určitým způsobem poškozen.

2.1 Extrakce charakteristických rysů

Extrakce charakteristických rysů je základem obrazového vyhledávání. Hlavní charakteristické rysy obrazu jsou barva, struktura a tvar obrazu.

2.1.1 Barva

Barva je jeden z nejpoužívanějších charakteristických vizuálních rysů v obrazovém vyhledávání, nezávislých na rozměrech a orientaci obrazu. K reprezentaci barvy se nejčastěji používá barevný histogram. Histogram je graf, udávající rozložení jednotlivých hodnot barevných složek v obrazu. Většinou se jedná o tři-složkový prostorový systém RGB, kde se barvy skládají ze tří základních složek: červené (**R**ed), zelené (**G**reen) a modré (**B**lue). Vzniká tak trojrozměrný prostor barev reprezentovaný jednotkovou krychlí, což je možno vidět na obrázku 1.



Obrázek 1: Prostorová krychle RGB systému [14]

2.1.2 Struktura - vazba

Struktura charakterizuje vizuální vzory. Je to přirozená vlastnost obrazu, která obsahuje důležitou informaci o strukturálním uspořádání mezi objektem v obrazu a prostředím, které ho obklopuje. Tato vlastnost dobře charakterizuje např. Fourierovo výkonové spektrum.

2.1.3 Tvar

V obrazovém vyhledávání se někdy požaduje, aby reprezentace tvaru obrazu byla invariantní vůči natočení a změně velikosti. Při obrazovém vyhledávání podle tvaru objektu v obrazu se úspěšně používají metody invariantních momentů a Fourierovy transformace.

2.2 Vysokodimensionální indexování

Při obrazovém vyhledávání v rozsáhlých obrazových databázích, kde jsou jednotlivé obrazy definovány jako vektory vysokodimensionálního prostoru, je účinné použít multidimensionálních indexovacích technik. Než se však přistoupí přímo k indexování, je zapotřebí snížení dimenze základních vektorů obrazů databáze. Ke snížení dimenze se v poslední době využívá singulárního rozkladu matic (viz kapitola 7.1.2). K indexování se pak použije např. kosinová míra podobnosti (viz kapitola 7.1.3). Vysokodimensionálního indexování využívá např. metoda LSI (*Latent Semantic Indexing*)

2.3 Systémy pro obrazové vyhledávání

Již od počátku devadesátých let 20. století se stalo obrazové vyhledávání velice aktivní výzkumnou oblastí. V následujícím přehledu budou představeny základní systémy pro obrazové vyhledávání.

- **QBIC** – Systém pracuje na principu geometrických invariantních momentů.
- **Virage** – Systém pracuje na vyhledávání obrazu pomocí barvy, barevného rozložení a struktury nebo kombinaci těchto parametrů.
- **RetrievalWare** – Systém pracuje na principu neuronových sítí. K vyhodnocení používá barvu, tvar, strukturu, jas a barevné rozložení obrazu.
- **Photobook** – Tento systém se snaží přiblížit lidskému subjektivnímu hodnocení obrazu.

- **VisualSEEK a WebSEEK** – Systém vyhledávání pomocí indexování v binárním stromu.
-
- **Netra** – Využívá k vyhodnocování barvu, strukturu a tvar obrazu.
-
- **MARS (Multimedia Analysis and Retrieval system)** – Je odlišný od všech výše zmíněných systémů. Využívá integrace databázového řízení, informačního vyhledávání, indexování a integrování člověka a počítače. Cílem není nalezení jednoho „nejlepšího“ řešení, ale jak organizovat vizuální proměnné do smysluplné vyhledávací architektury, která by se dynamicky adaptovala na různé aplikace.

2.4 Směry výzkumu v budoucnu

Přes veškerou snahu nalézt plně automatizovaný vyhledávací systém pro obrazové databáze, je člověk stále nezastupitelnou složkou v počítačovém vyhledávacím procesu. Člověk je konečným článkem systémů pro obrazové vyhledávání. Proto je ve výzkumu v této oblasti trend zkoumat spolupráci mezi člověkem a počítačem. Je nutné prozkoumat z psychologického hlediska lidské vnímání obsahu obrazu. Toto téma získává v posledních letech zvýšené pozornosti, zaměřené především na výzkum subjektivního lidského vnímání obrazu a integraci „lidského modelu“ do systému obrazového vyhledávání.

3 Popis problému

K dispozici byla databáze 454 obrázků otisků podešví bot, které byly převedeny na stejnou velikost. Aby vyhledávání v této referenční databázi bylo realizovatelné, musí mít obrázky databáze a obrázek dotazu (vyhledávaný obrázek) shodné rozměry. Obrázky byli v 8 bitové hloubce, tj. v 256 úrovních šedi.

Úkolem této práce je nalézt optimální metodu pro vyhledávání v obrazové databázi otisků podešví bot. Tzn. vytrdit z původní databáze 454 obrázků subdatabázi nejpodobnějších obrázků odpovídajících obrázku dotazu, seřazených sestupně podle hodnoty míry podobnosti mezi obrázkem databáze a obrázkem dotazu. V této subdatabázi se musí nacházet protějšek hledaného obrázku z referenční databáze. Snahou bude, aby takto seřazená subdatabáze měla co nejvyšší rozsah a hledaný obrázek (resp. jeho protějšek z původní databáze) se v ní nacházel na co nejvyšším místě.

Pro vyhledávání byly použité tři metody:

1. **Metoda LSI (*Latent Semantic Indexing*)**. Databáze obrázků se předpracuje do předem navržené struktury - indexu. Cílem indexování je umožnit rychlé vyhodnocení uživatelského dotazu pomocí indexu.
2. **Metoda DFT (*Discrete Fourier Transform*)**. K vyhledávání používá dvojrozměrnou Fourierovu transformaci. Dvojrozměrná Fourierova transformace umožní převést rozložení obrazových intenzit $f(x,y)$ vyhodnocovaného obrazu na obraz prostorových frekvencí $F(f_x, f_y)$, tzv. obrazové spektrum.
3. **Metoda centrálních momentů**. Je založena na geometrických invariantních momentech. Tyto momenty produkují soubor hlavních vektorů, které jsou invariantní vůči posunutí, natočení a změně velikosti obrazu.

Pro metody LSI a DFT budou provedeny tři experimenty

- a) Vyhledávaný obrázek bude natáčen v intervalu $\pm 10^\circ$ od svislé osy po kroku 1° .
- b) Kvalita vyhledávaného obrázku bude degradována částečným, subjektivním odmazáním stopy a přidáním Gaussovského bílého šumu (dále jen GBŠ), aby se obraz co nejvíce přibližoval reálnému otisku. Současně bude vyhledávaný obrázek natáčen na intervalu $\pm 10^\circ$ od svislé osy po kroku 1° .

- c) Budou použity reálné otisky stopy (otisk sejmutý přímo z podlahy), pro vyhledávání v referenční databázi.

Při vyhledávání budou sledovány tyto parametry:

- Úspěšnost vyhledávání – na jakém místě se vyhledávaný obrázek nachází ve vytřížené subdatabázi nejpodobnějších obrázků, odpovídajících obrázku dotazu.
- Citlivost metody na natočení obrázku dotazu.
- Citlivost metody na poškození obrázku dotazu.
- Časová náročnost vyhledávání.

U metody centrálních momentů bude zkoumáno využití této metody pro klasifikaci referenční databáze pomocí klasifikačního stromu. Zde bude sledováno, jestli se vyhledávaný obrázek nachází ve stejné větvi stromu, jako jeho protějšek z referenční databáze.

4 Shrnutí současného stavu

Práce [2] se věnuje problému využití LSI (*Latent Semantic Indexing*) a M-stromu při indexování a vyhledávání databáze 730 obrázků budov. Tato databáze byla složena ze skupin po pěti obrázcích, zobrazujících stejnou budovu z různých úhlů pohled. V této práci autoři prezentují vytváření vektorů jednotlivých obrázků, kde je dále celá databáze obrázků prezentována jako matice A , jejíž sloupce představují jednotlivé obrázky databáze. Pomocí modelu LSI (resp. singulárního rozkladu) dochází k sémantickému indexování konceptů. Vyhledávání bylo uskutečněno pomocí M-stromu. Autoři vyhledávali 4 nejpodobnější obrázky k obrázku dotazu. Pokud jeden ze čtyř nalezených obrázků odpovídal hledanému obrázku, byla úspěšnost vyhledávání 25%, pokud odpovídaly dva, byla úspěšnost vyhledávání 50% atd. Autoři uvádějí průměrnou úspěšnost vyhledávání 43%. Singulární rozklad bude podrobněji popsán v kapitole 7.1.2

Automatizovanému vyhledávání otisků podešví bot pomocí metody diskrétní Fourierovy transformace se zabývají tvůrci v práci [12]. Autoři měli k dispozici referenční databázi 1276 obrázků otisků podešví bot. Obrázky v této databázi byly převedeny na stejnou velikost a oprahovány. Prahování digitálního obrazu bude popsáno v kapitole 5.2. Dále byly pro každou stopu spočítána Fourierova výkonová spektra, ze kterých byly odfiltrovány nízké resp. vysoké frekvence. Pomocí korelačního koeficientu byly takto upravená spektra porovnávána se spektrem hledané stopy, upraveným stejným postupem. Korelační koeficient bude popsán v kapitole 5.3. Z tohoto experimentu došli autoři k závěru, že vyhledávání je citlivé na použité velikosti obrázků z referenční databáze a obrázku dotazu. Dále bylo zjištěno, že se důležité informace o vzoru podešve skrývají v nižší frekvenční části Fourierova spektra.

Práce zabývající se rozpoznáním objektu pomocí metody centrálních momentů zpracoval Mohamed Rizon a kol. [1] z Univerzity Kejuruteraan Utara v Malajsii. Tato práce hledá řešení pro automatizaci sklizení kokosových ořechů, které jsou nepostradatelnou komoditou v Malajské ekonomice. Autoři se snaží pomocí centrálních momentů rozlišit v databázi 50 obrázků kokosový ořech od vějířovitých listů palmy. K rozpoznávání objektu v obraze při tomto experimentu nejlépe vyhovoval první centrální moment, u kterého byly rozdíly hodnot mezi centrálními momenty kokosového ořechu a listy palmy nejvýraznější. Udávaná úspěšnost rozpoznání je 70 – 90%.

5 Digitální obraz

Obrazem budeme rozumět digitální obraz s plošnou (dvojměrnou) prezentací reálné trojrozměrné scény. Příkladem je obraz na sítnici lidského oka nebo obraz viděný TV kamerou. Matematickým modelem obrazu může být spojitá funkce $f(x,y)$ dvou argumentů souřadnic v rovině. Funkce $f(x,y)$ se obvykle nazývá *obrazová funkce*. Hodnotou obrazové funkce je nejčastěji jas resp. intenzita jasu, protože je veličinou měřenou většinou čidel pro vstup obrazu. Jas je veličina, která souhrnně vyjadřuje vlastnosti obrazového signálu způsobem, který odpovídá jeho vnímání člověkem. Jas závisí na typu světelných zdrojů, jejich poloze, intenzitě, poloze pozorovatele, lokální geometrii povrchu a odrazivosti povrchu.

Hodnota charakterizující bod v černobílém binárním obraze, může mít jen dvě úrovně (0 – černá, 1 – bílá). V monochromatickém obraze vyjadřuje tato hodnota úroveň šedi (jasu) bodu. V složitějším případě se pracuje s barevným obrazem, který je reprezentovaný vektorem barvy obrazového bodu. Pro barevné obrazy je většinou vektor barvy bodu daný třemi složkami RGB (R – červená, G – zelená, B – modrá). Každé dvojici plošných souřadnic (x,y) odpovídá vektor hodnot – např. jasů pro jednotlivé barevné složky obrazu [7, 8].

V této práci se budeme dále zabývat výhradně monochromatickými obrazy v 8 bitové hloubce, v 256 úrovních šedi ($R = G = B$).

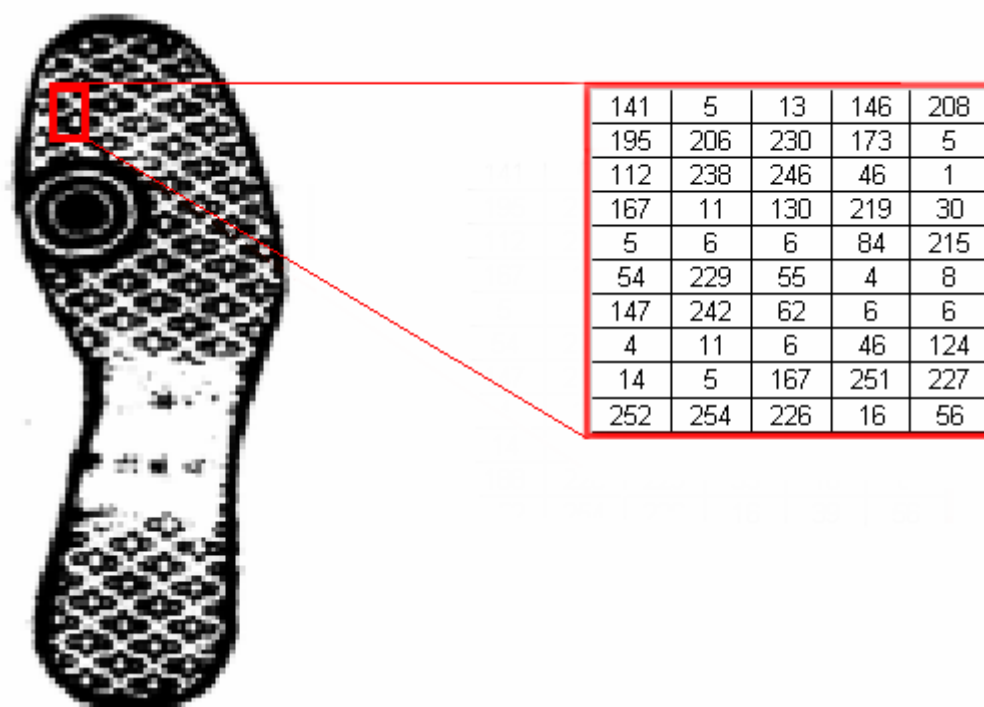
5.1 Obrazová struktura

Matice je nejobvyklejší datovou strukturou pro reprezentaci obrazu, data bývají přímo výstupem snímacího zařízení. Prvky matice jsou celá čísla, která odpovídají funkční hodnotě šedi v příslušném bodě vzorkovací čtvercové mřížky $f(x,y)$. Na obrázku 2 je zobrazený obecný tvar obrazové matice o velikosti $m \times n$ kde m značí počet řádků a n počet sloupců.

$$f(x, y) = \begin{pmatrix} f(0,0) & \dots & f(0,n-1) \\ \vdots & \ddots & \vdots \\ f(m-1,0) & \dots & f(m-1,n-1) \end{pmatrix}$$

Obrázek 2: Obecný tvar obrazové matice.

Na obrázku 3 je znázorněn příklad digitalizovaného obrázku otisku podešve boty o velikosti 150×70 pixelů v 256 úrovních šedi, kde je vlevo v červeném obdélníku znázorněn výběr určitého segmentu stopy č.23 o velikosti 10×5 pixelů a vpravo jsou odpovídající hodnoty úrovní šedi.



Obrázek 3: Znáznornění digitalizace na stopě č.23.

5.2 Prahování obrazu

Prahování je jedna z nejstarších a nejjednodušších metod segmentace obrazu a patří také k jedné z nejrychlejších. Ve většině případů vystupují objekty z pozadí, čímž je myšleno, že jasové (barevné) hodnoty prvků obrazové funkce (pixelů) objektů jsou odlišné od pozadí. Na tomto předpokladu je založená metoda prahování. Jejím principem je najít vhodný práh (hodnotu jasu), který by oddělil objekty do pozadí. Výsledkem tohoto (jedno-prahového) procesu je obraz v binárním tvaru, ve kterém mají hodnotu 0 ty prvky obrazové funkce, které patří objektům a hodnotu 1 ty, které patří pozadí [9] .

5.3 Korelace obrazů

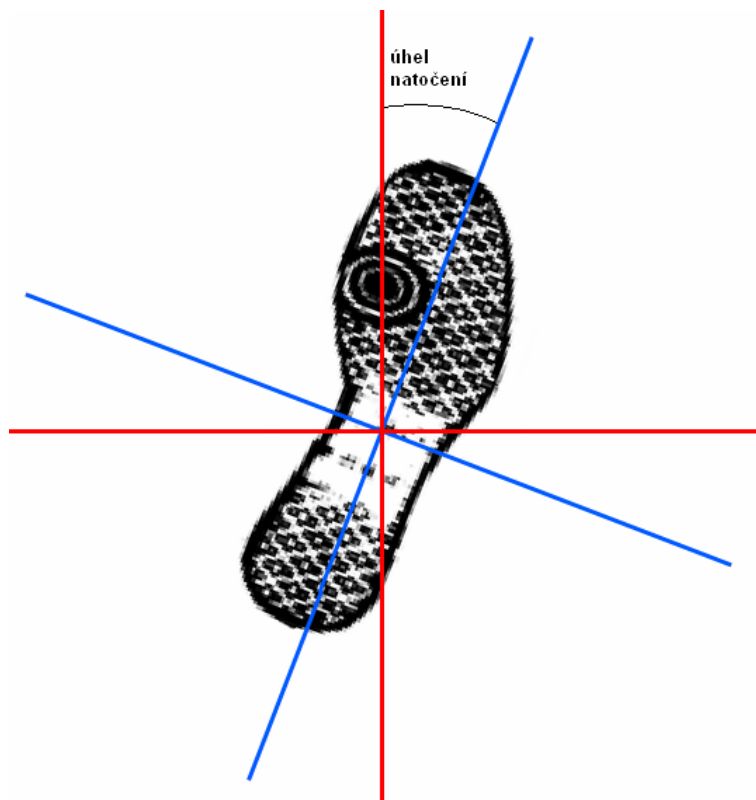
Jak bylo výše zmíněno, lze digitální obraz popsat tzv. obrazovou maticí o velikosti $m \times n$, kde m značí počet řádků a n počet sloupců. Necht' A a B jsou obrazové matice stejné velikosti $m \times n$. Pak korelační koeficient mezi těmito maticemi je definován jako

$$r = \frac{\sum_i \sum_j (A_{ij} - \bar{A})(B_{ij} - \bar{B})}{\sqrt{\left(\sum_i \sum_j (A_{ij} - \bar{A})^2 \right) \left(\sum_i \sum_j (B_{ij} - \bar{B})^2 \right)}}, \quad (1)$$

kde \bar{A} je aritmetický průměr hodnot matice A a \bar{B} je aritmetický průměr hodnot matice B .

5.4 Natočení obrazu

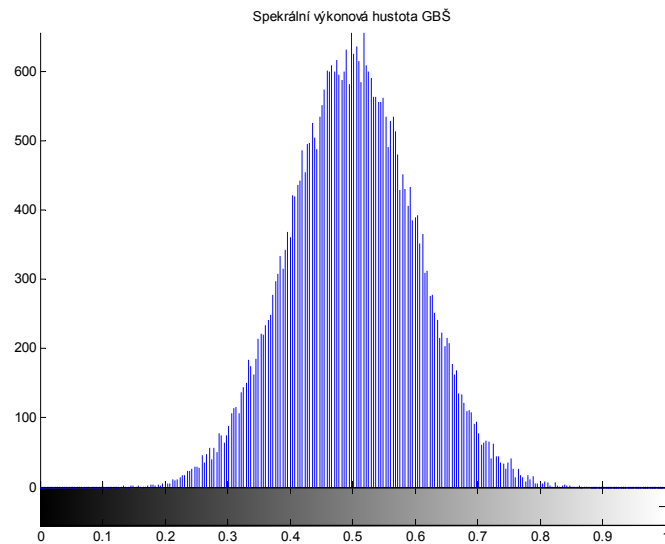
Natočením obrazu budeme rozumět natočení svislé osy obrazu od svislé osy zavedeného systému. Úhel, který svírají tyto dvě osy bude dále nazýván úhel natočení. Na obrázku 4 je znázorněn příklad takového natočení, kde červené osy jsou osy zavedeného systému a modré osy jsou osy obrázku. Natočení doleva bude dále bráno jako kladné a doprava jako záporné.



Obrázek 4: Znáznornění úhlu natočení obrázku (záporné natočení).

5.5 Gaussovský, bílý šum

Šum je obecně nežádoucí změna obrazu. Tato změna degraduje kvalitu obrazu. Při experimentech bude používán Gaussovský, bílý šum. Funkci vyvolání GBŠ s konstantní hodnotou průměru a rozptylu v MATLABu vykonává příkaz *imnoise* (*I*, 'gaussian', *w*, *g*), kde je *w* průměrná hodnota, *g* rozptyl a *I* obrázek, který má být zašuměn. Na obrázku 5 je znázorněna spektrální výkonová hustota GBŠ ($w = 0,5$; $g = 0,01$).



Obrázek 5: *Spektrální výkonová hustota bílého šumu [17]*

Při provedených experimentech budou uvažovány dvě intenzity GBŠ a to:

- $w = g = 0,2$ – střední intenzita šumu
- $w = g = 0,1$ – nízká intenzita šumu

6 Rozdělení databáze do 9-ti podskupin

V praxi se velmi často stává, že reálné otisky podešví bot nejsou úplné, ale podešev zanechá jen částečný otisk. Aby bylo realizovatelné vyhledávání i neúplných otisků, bude potřeba referenční databázi rozdělit do těchto podskupin :

Full – celá stopa, HalfLeft - levá polovina stopy, HalfRight - pravá polovina stopy, HalfTop - horní polovina stopy, HalfBottom - dolní polovina stopy, LeftBottom - levá dolní čtvrtina stopy, RightBottom - pravá dolní čtvrtina stopy, LeftTop - levá horní čtvrtina stopy, RightTop - pravá horní čtvrtina stopy, podle určitého dělicího bodu (*Origin pointu* - dále jen OP).

6.1 Převzorkování obrazů databáze na stejnou velikost :

Jelikož v referenční databázi nemají všechny obrázky stejnou velikost, bylo nutné je převzorkovat na stejnou velikost. To bylo provedeno tak, že se šířka všech obrázků resamplovala na hodnotu šířky nejširšího obrázku a výška se resamplovala na hodnotu výšky nejvyššího obrázku z původní databáze. Velikost všech obrázků stop byla poté převzorkována na velikost 1200×564 pixelů. Pro zrychlení výpočtů bylo nutné obrázky databáze zmenšit na konečných 150×70 pixelů.

6.2 Nalezení Origin Pointu (dělicího bodu)

Výše popsaným převzorkováním originální databáze vznikla databáze, kterou bylo pro další potřeby vyhledávání ještě nutno rozdělit do výše zmíněných 9-ti podskupin. Aby mohly být obrazy rozděleny do těchto podskupin, bylo nutné určit bod, podle kterého se rozdělení bude provádět, tzv. *Origin point*. Algoritmus pro nalezení OP byl navržen podle následujících kroků:

1. Nalezení prvního krajního sloupce, kde zasahuje objekt souřadnice s_1 .
2. Nalezení posledního krajního sloupce, kde zasahuje objekt souřadnice s_2 .
3. Nalezení prvního krajního řádku, kde zasahuje objekt souřadnice r_1 .
4. Nalezení posledního krajního řádku, kde zasahuje objekt souřadnice r_2 .

Z těchto bodů se spočítaly souřadnice počátku pro všechny databázové obrázky

$$sloupec = \frac{(s_1 - s_2)}{2}, \quad (2)$$

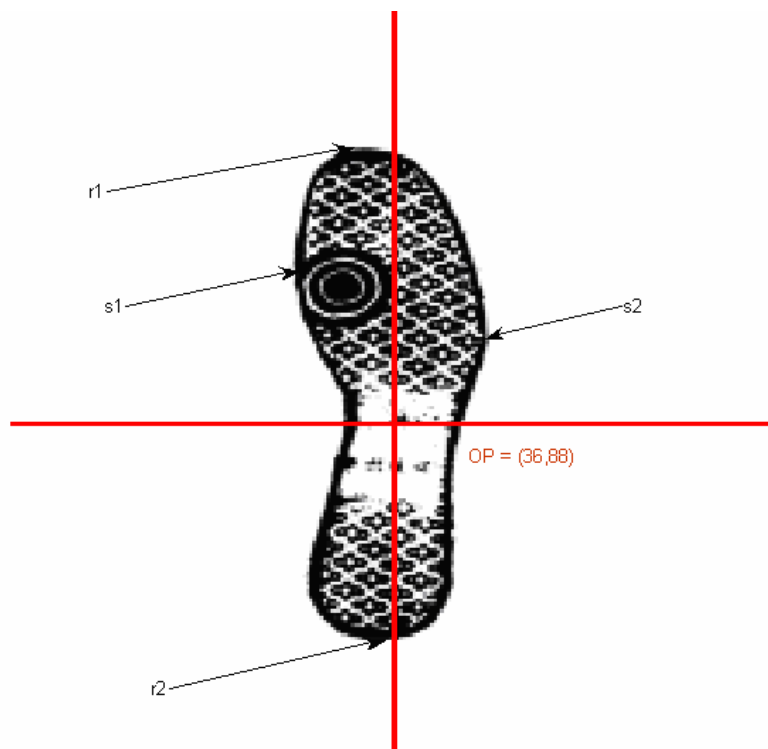
$$řádek = 0,58(r_1 - r_2). \quad (3)$$

Souřadnice OP se zjistily zprůměrováním souřadnic počátků všech stop databáze

$$(r, s) = \left(\frac{1}{PS} \sum_{i=1}^n řádek_{(i)}, \frac{1}{PS} \sum_{i=1}^n sloupec_{(i)} \right), \quad (4)$$

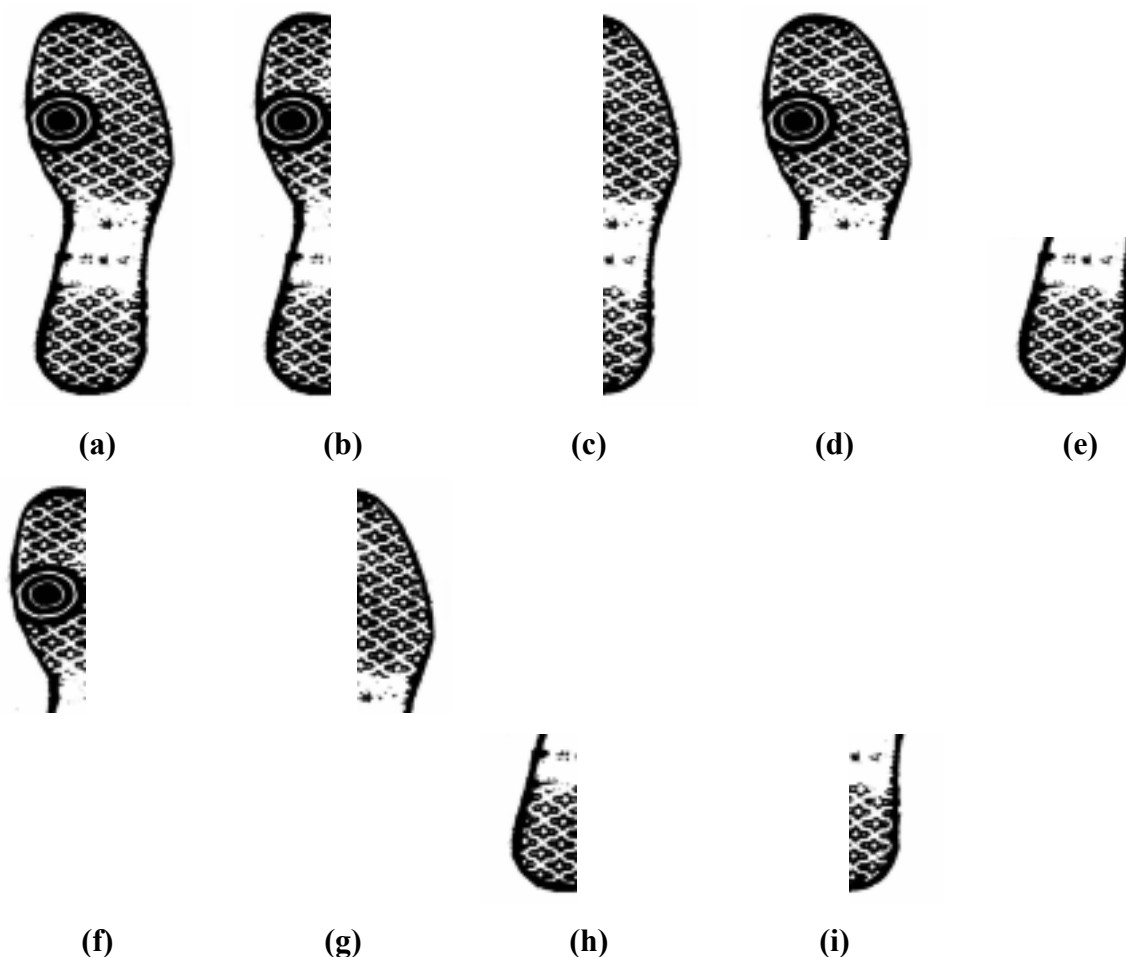
kde je PS - počet stop databáze, $řádek_{(i)}$ - souřadnice počátku i-tého obrázku pro řádek (osa x), $sloupec_{(i)}$ - souřadnice počátku i-tého obrázku pro sloupec (osa y), r - souřadnice Origin pointu pro řádek (osa x), s - souřadnice Origin pointu pro sloupec (osa y).

Souřadnice spočítaného OP byly $r = 36$ a $s = 88$, $OP = (36,88)$.



Obrázek 6: Znáznornění souřadnic počátku a Origin pointu na stopě č.23.

Po nalezení Origin pointu pomocí souřadnic počátku všech obrázků databáze, jak je znázorněno na obrázku 6, byly obrázky databáze podle tohoto bodu rozděleny do 9-ti podskupin, jak je vyobrazeno na obrázku 7. Velikost obrázků v každé podskupině v pixelech je zanesena do tabulky 1.



Obrázek 7: Názorná ukázka podskupin databáze pro stopu č. 23. (a) Full – celá, (b) HalfLeft - levá polovina, (c) HalfRight - pravá polovina, (d) HalfTop - horní polovina, (e) HalfBottom - dolní polovina, (f) LeftTop - levá horní čtvrtina, (g) RightTop - pravá horní čtvrtina, (h) LeftBottom - levá dolní čtvrtina, (i) RightBottom - pravá dolní čtvrtina.

Tabulka 1: Zobrazení velikosti obrázků v každé podskupině.

podskupina	velikost obrázku	
<i>Full</i>	<i>150 x 70</i>	<i>pixelů</i>
<i>HalfLeft</i>	<i>36 x 150</i>	<i>pixelů</i>
<i>HalfRight</i>	<i>35 x 150</i>	<i>pixelů</i>
<i>HalfTop</i>	<i>70 x 88</i>	<i>pixelů</i>
<i>HalfBottom</i>	<i>70 x 63</i>	<i>pixelů</i>
<i>LeftBottom</i>	<i>36 x 63</i>	<i>pixelů</i>
<i>RightBottom</i>	<i>35 x 63</i>	<i>pixelů</i>
<i>LeftTop</i>	<i>36 x 88</i>	<i>pixelů</i>
<i>RightTop</i>	<i>35 x 88</i>	<i>pixelů</i>

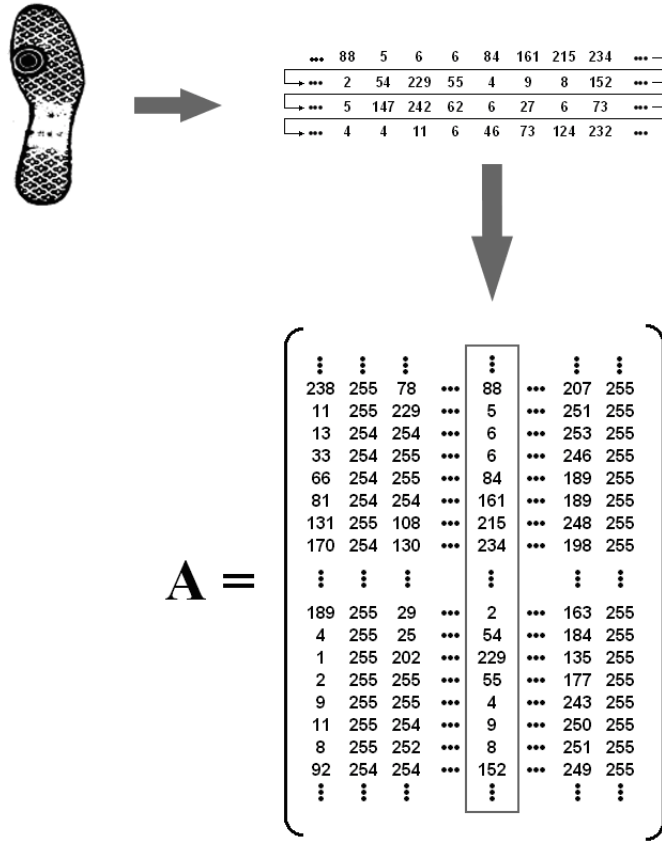
7 TEORETICKÁ ČÁST

7.1 Latent Semantic Indexing (LSI)

Pokud chceme jednoduše a efektivně vyhledávat v nějaké databázi multimediálních dat (v tomto případě databázi obrázků otisků podešví bot), musíme tato multimediální data předem nějakým vhodným způsobem definovat a organizovat. Jednotlivé multimediální dokumenty (obrázky) lze definovat jako vektory vysokodimenzionálního prostoru. Pomocí sekvenčního vyhledávání, je celá databáze postupně načtena a za pomoci určité funkce podobnosti se vzájemně porovnávají vektory dokumentů databáze s vektorem dokumentu dotazu. Tímto způsobem lze v prostoru vyhledávat. Tento způsob vyhledávání je při rozsáhlejší databázi dokumentů velmi náročný, proto byly navrženy metody, které nejdříve databázi předzpracují do předem navržené struktury – indexu. Cílem takového indexování je umožnit rychlé vyhodnocení dotazu pomocí indexu [3].

7.1.1 Extrakce vektorů vlastností pomocí metody LSI

Nejdříve byl každý obrázek o velikosti $m \times n$ (resp. jasy/stupně šedi všech jeho pixelů) převeden do *jasového vektoru* obrázku. Bylo provedeno zcela triviálně vektorizací obrazové matice obrázku (jednotlivé řádky matice se "naskládaly" za sebe). Tento řádkový vektor byl transponován na sloupcový vektor o velikosti $((m * n) \times 1)$. Takto zhotovené jasové vektory byly uloženy do *matice dokumentů* A , tj. každý sloupec matice znázorňuje jasový vektor obrázku dané stopy z referenční databáze. $A = ((m * n) \times 454)$ [3], jak je vidět na obrázku 8.



Obrázek 8: Transformace obrázku do jasového vektoru a jeho umístění v matici dokumentů A

7.1.2 Singulární rozklad matice dokumentů A

Věta: Reálna matice A typu $((m * n) \times ps)$; $m * n \geq ps$. Pak existuje matice U typu $((m * n) \times ps)$, diagonální matice S řádu ps s nezápornými diagonálními prvky a čtvercová matice V řádu ps tak, že

$$A_{((m.n) \times ps)} = U_{((m.n) \times ps)} S_{(ps \times ps)} V_{(ps \times ps)}^T \quad (5)$$

Důkaz viz [4].

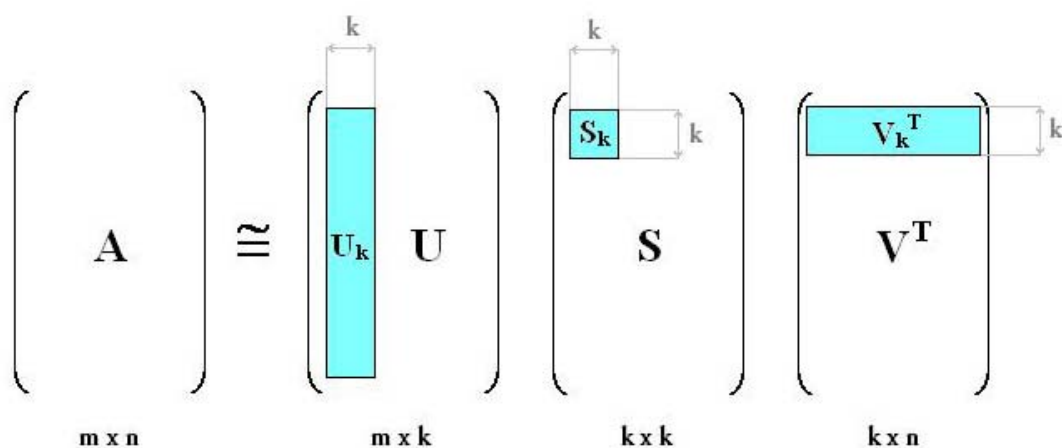
Sloupce matice U (tzv. *vektory konceptů*) lze interpretovat jako jednotlivá témata přítomná v databázi obrázků. Vektory konceptů jsou uspořádány podle důležitosti, tj. podle hodnot singulárních čísel uložených sestupně v S , podle toho v jaké míře a v kolika dokumentech se vyskytují. Tím je určeno, které koncepty jsou vzhledem k databázi sémanticky důležité =>

latentní – skryté sémantické indexování. Nedůležité koncepty působí jako statistický šum. Sloupce matice SV^T obsahují vektory dokumentů tzv. *vektory pseudo-dokumentů*. Jeden vektor pseudo-dokumentů lze vyjádřit jako lineární kombinaci všech nalezených vektorů konceptů, tj. že příslušný dokument obsahuje v nějaké míře každý z konceptů.

Vzhledem k tomu, že lze považovat za sémanticky důležité jen prvních k konceptů, je vztah možné aproximovat jako

$$A_{((m.n) \times ps)} = U_{((m.n) \times k)} S_{(k \times k)} V_k^T_{(k \times ps)}, \quad (6)$$

kde v matici U_k je prvních k nejdůležitějších vektorů konceptů, v matici S_k příslušná singulární čísla a ve $S_k V_k^T$ vektory pseudo-dokumentů, vyjádřené pomocí prvních k vektorů konceptů, které je znázorněno na obrázku 9. Singulární rozklad matice A (dále jen SVD rozklad) nám promítne původní m -rozměrné vektory dokumentů do prostoru dimenze k ($k \ll m$).



Obrázek 9: Aproximace singulárního rozkladu matice A

7.1.3 Kosinová míra podobnosti

Předpokládejme, že pro indexaci všech dokumentů v databázi bylo použito celkem n různých klíčových vektorů t_1, \dots, t_n , potom každý dokument D_i ze souboru dokumentů D je reprezentován vektorem $D_i = (w_{i1}, w_{i2}, \dots, w_{in})$, kde $w_{ij} \in R$ a w_{ij} jsou váhy náležející

klíčovému slovu t_j v dokumentu D_i . Váha w_{ij} určuje důležitost jednotlivých klíčových vektorů pro identifikaci dokumentu. Pokud je váha rovna nule, pak představuje nejnižší důležitost. Je-li váha rovna jedné, má nejvyšší důležitost. Soubor dokumentů D je ve vektorovém modelu popsán maticí

$$D = \begin{pmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{m1} & \cdots & w_{mn} \end{pmatrix}$$

ve které i -tý řádek odpovídá i -tému dokumentu a j -tý sloupec odpovídá j -tému klíčovému vektoru. Výraz dotazu Q ve vektorovém modelu je možné formulovat jako n -místný vektor vah $Q = (q_1, q_2, \dots, q_n)$, kde $q_j \in R$. Na základě dotazu Q lze pro každý dokument D_i spočítat tzv. *koeficient podobnosti*. Tento koeficient si lze představit jako „podobnost“ vektoru dokumentu s vektorem dotazu ve vektorovém prostoru R^n . Výpočet podobnosti dokumentu D_i vzhledem k dotazu Q je definován vzorcem

$$Sim(Q, D_i) = \frac{\sum_{k=1}^n (q_k w_{ik})}{\sqrt{\sum_{k=1}^n (w_{ik})^2 \sum_{k=1}^n (q_k)^2}}. \quad (7)$$

Tento koeficient podobnosti se nazývá *kosinová míra podobnosti* a udává vzdálenost mezi dvěma vektory [10, 11].

7.1.4 Vyhledávání vektoru v matici dokumentů

Vyhledávaný vektor (*vektor dotazu* q) je porovnáván v matici dokumentů na základě kosinové míry podobnosti. Aby bylo možné porovnávat vektory pseudo-dokumentů s vektorem dotazu, je potřeba zkonstruovat z vyhledávaného vektoru jeho projekci do U_k :

$$qc = q' U_k S_k^{-1}, \quad (8)$$

$$míra\ podobnost(i) = \frac{qc * V(i,:)'}{norm(qc) * norm(V(i,:))}, \quad (9)$$

kde je $norm(X)$ je euklidovská vzdálenost vektoru X .

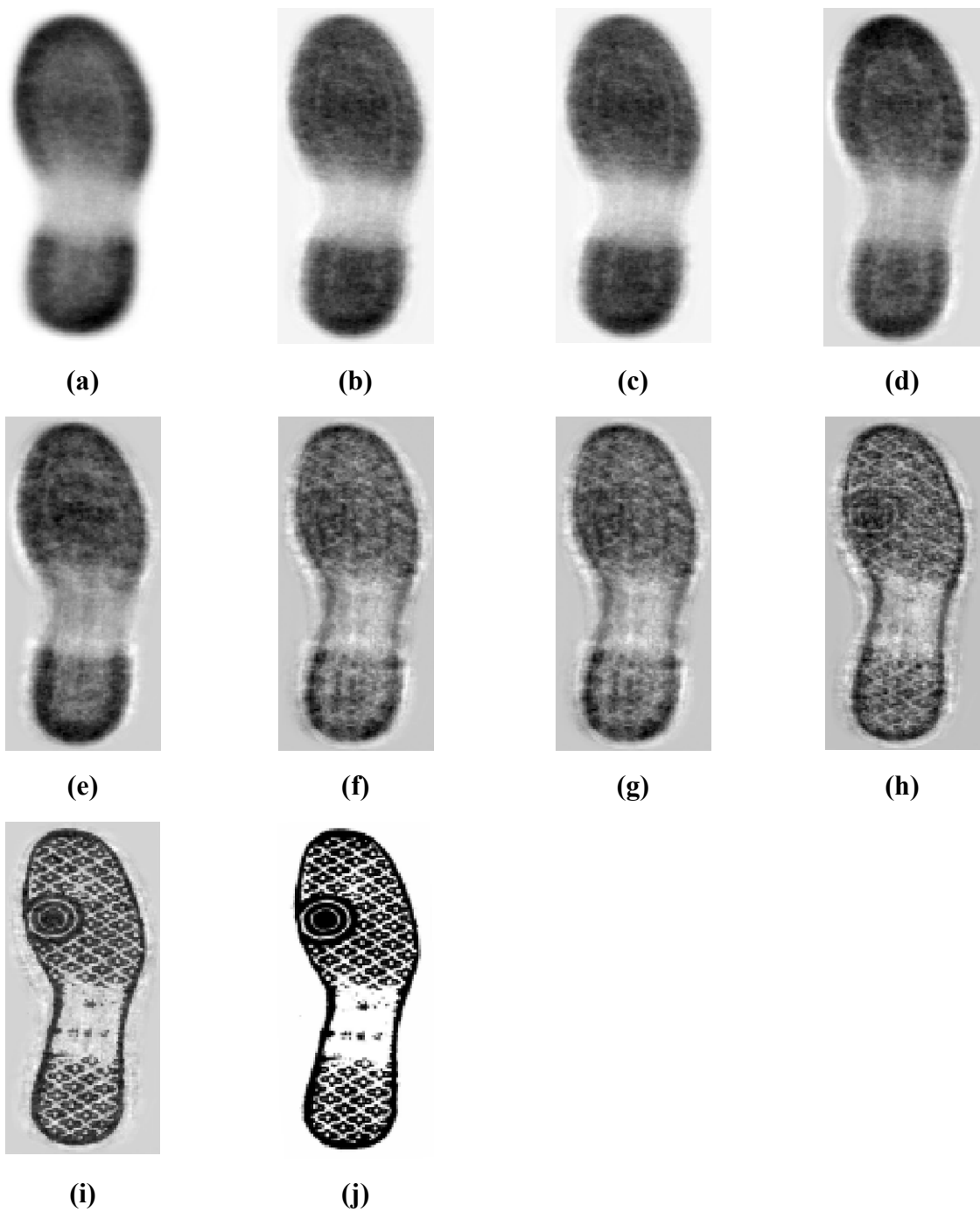
7.1.5 Použití LSI pro databázi obrázků

SVD rozklad lze aplikovat na libovolné matice. Lze je použít i při vyhledávání v jakékoli multimediální databázi, v tomto případě v databázi obrázků otisků podešví bot, kterou lze reprezentovat sadou vektorů stejné dimenze. Jedinou podmínkou sestavení matice A je, aby jednotlivé vektory vlastností dokumentů tvořili sloupce matice A , tj. každý sloupec matice A představuje vektor vlastností daného obrázku. V našem případě byla matice A sestavena, jak bylo výše popsáno a vyobrazeno na obrázku 8.

7.1.6 Rekonstrukce obrázků

Při SVD rozkladu je též zajímavá vizualizace vektorů konceptů (*báze U*), které představují jakési "bázové obrázky". Platí tedy, že vhodnou kombinací bázových obrázků dostaneme libovolný obrázek z databáze, tzn. čím vyšší hodnota k , tím přesnější pak bude rekonstrukce obrázku. Na obrázku 10 je ilustrováno, jak velké k je dostačující pro popis obsahu obrázku. Při malém k je rekonstrukce velmi nedokonalá, což je znázorněno na obrázku 10(a) – (e), ale i u takto hrubě reprezentovaných obrázků, lze rozeznat tvary původních stop. Je třeba si však uvědomit, že např. při $k = 32$ (viz obrázek 10(f)), kdy už je na obrázku viditelný tvar původní stopy, je obrázek složen pouze z 32 bázových obrázků. Tj. že místo 10500 hodnot jasů pixelů (obrázek z podskupiny *Full* - 150x70 pixelů) je potřeba pouze 32 "vah" konceptů. Pro $k = 256$ je již rekonstrukce na takové úrovni, že je možné stopu jednoznačně rozpoznat. Toto ilustruje obrázek 10(i). Na obrázku 10(j) je vidět původní obrázek stopy z referenční databáze, resp. obrázek, který byl zrekonstruován po plném rozkladu $A = USV^T$ (kde k je rovno hodnotě matice U^T).

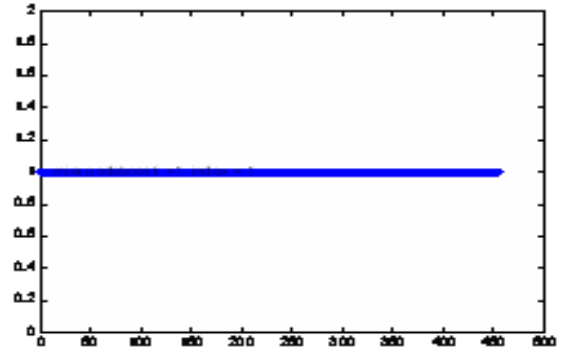
Z uvedeného příkladu je zřejmé, že pokud budeme vyhledávat podle hrubých rysů, je vhodné použít nízkou hodnotu k . Naopak pokud budeme vyhledávat podle detailů, musíme použít vyšší hodnotu k . Zajímavé je podívat se na to, jak ovlivňuje velikost použitého k hodnoty míry podobnosti mezi hledaným obrázkem a jednotlivými obrázky databáze. To je možno vidět na obrázku 11.



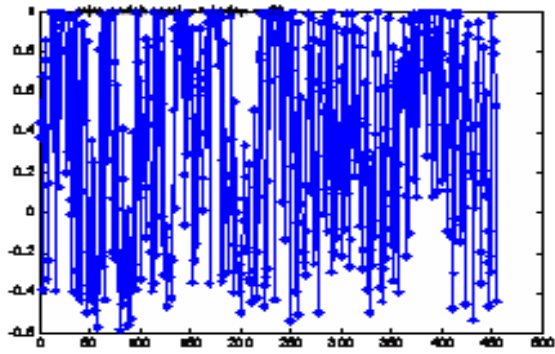
Obrázek 10: Rekonstrukce obrázku z SVD rozkladu. (a) Plný rozklad $A = USV^T$, $k = \text{hodnosti matice } U^T$, (b) $k = 1$, (c) $k = 2$, (d) $k = 4$, (e) $k = 8$, (f) $k = 16$, (g) $k = 32$, (h) $k = 64$, (i) $k = 128$, (j) $k = 256$.



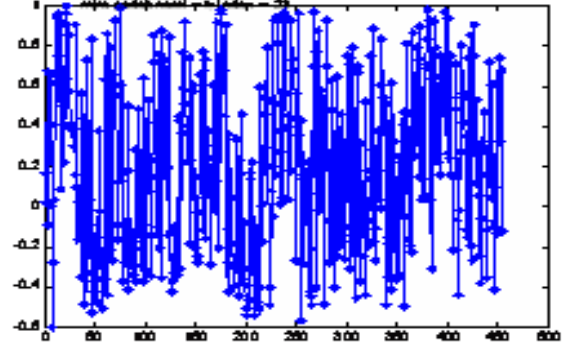
(a)



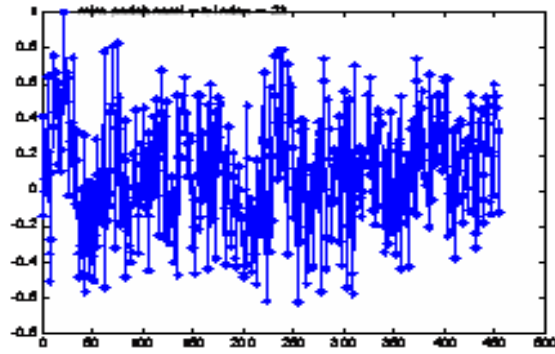
(b)



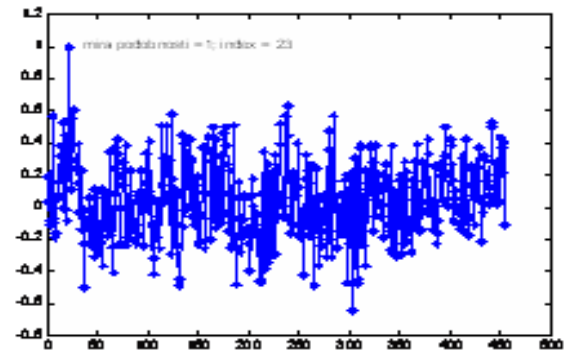
(c)



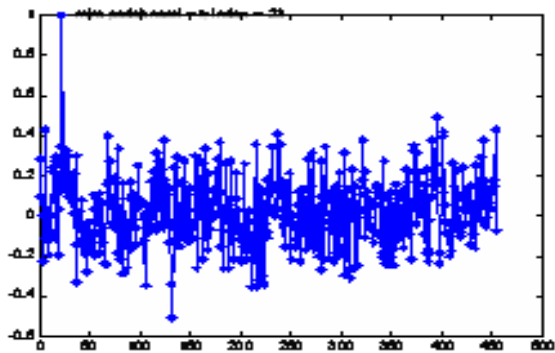
(d)



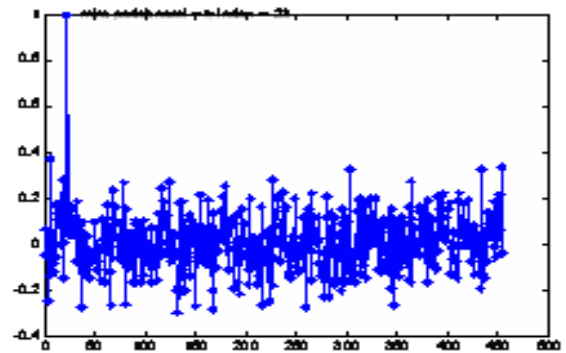
(e)



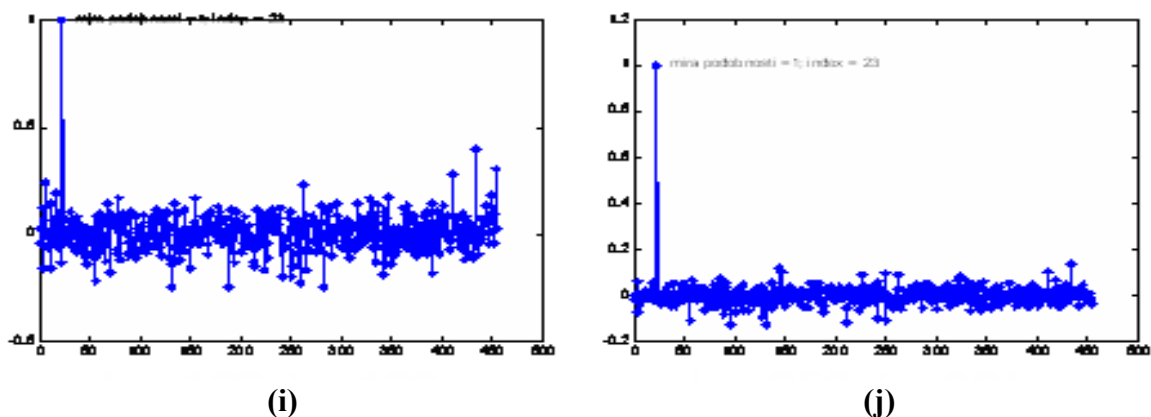
(f)



(g)



(h)



Obrázek 11: Závislost hodnot míry podobnosti na velikosti použitého k . (a) Vyhledávaný obrázek - stopa č.23, (b) $k = 1$, (c) $k = 2$, (d) $k = 4$, (e) $k = 8$, (f) $k = 16$, (g) $k = 32$, (h) $k = 64$, (i) $k = 128$, (j) $k = 256$.

Na obrázku 11 je vidět, jak se rozptýl spočítaných hodnot míry podobnosti mezi hledaným obrázkem a jednotlivými obrázky databáze zužuje. Naopak hodnota míry podobnosti mezi hledaným obrázkem a jeho protějškem z referenční databáze je z tohoto intervalu se vzrůstajícím k více vybočující. Na obrázku 11(e) je hodnota $k = 8$, která je dostatečně velká na to, aby hodnota míry podobnosti mezi hledaným obrázkem a jeho protějškem na první pohled znatelně vybočovala z intervalu rozptýlu spočítaných hodnot míry podobnosti obrázků databáze. Na obrázku 11(g), kdy se $k = 32$ je tento rozdíl zcela evidentní.

7.2 Diskrétní Fourierova transformace

V praxi je často výhodné používat harmonické funkce, neboť jsou snadno prakticky realizovatelné (resp. jejich imaginární či reálná část) a mají výhodné matematické vlastnosti. Ukazuje se, že za dosti širokých podmínek lze každou funkci vyjádřit jako součet či integraci harmonických funkcí, ovšem každé s jinou váhou a fázovým posuvem (zpravidla jsou obě hodnoty zahrnuty do komplexní váhové funkce). Váhová funkce tedy udává, jaké frekvence je nutné použít v superpozici, aby bylo možné z harmonických funkcí zpětně sestavit původní funkci. Právě tato váhová funkce (*spektrum*) bývá označována jako *Fourierova transformace* (FT). Definiční vzorec pro FT je integrálem a pro praktickou realizaci není příliš vhodný:

- Jeho analytické řešení existuje jen v omezeném počtu případů a je nutno jej tedy řešit numericky (tedy převodem *nekonečný integrál* \rightarrow *konečná sumace*).
- V případě počítačového zpracování nemáme spojitou funkci, ale jen její hodnoty v diskrétních vzorkovacích okamžicích.

Z těchto důvodů se definuje diskrétní Fourierova transformace (DFT), která je již polynomem a jejími vstupy a výstupy jsou posloupnosti hodnot. Nevýhodou této definice je značná časová náročnost jejího výpočtu, která roste s druhou mocninou délky vstupní posloupnosti (N^2). Proto byl vypracován algoritmus, který vychází z vlastností exponenciálních diskrétních funkcí a výrazně snižuje potřebnou dobu výpočtu. Tento algoritmus je zvykem nazývat rychlá Fourierova transformace (FFT – *Fast Fourier Transform*) [6]. Fourierova transformace se ukázala být účinnou metodou zpracování různých signálů. Často je využíváno její vlastnosti převodu konvoluce na násobení, což umožňuje u některých soustav zavést tzv. přenosovou (frekvenční) funkci, která vhodným způsobem charakterizuje dynamické vlastnosti soustavy. Metoda umožňuje provádět frekvenční filtraci, tedy odstraňovat ze signálu části s různými frekvencemi, což může např. snížit úroveň šumu v signálu. Operace ve frekvenční oblasti mohou upravovat obrazy takovým způsobem, aby např. došlo ke zvýraznění hran, k odstranění „proužkování“ či ke zvýraznění některých struktur v obraze. Výrazným uplatněním FT je také skutečnost, že mnohé fyzikální jevy mohou být aproximovány právě Fourierovou transformací.

7.2.1 2D diskrétní Fourierova transformace

Dvourozměrnou Fourierovu transformaci můžeme definovat jako

$$F(u, v) = \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} f(x, y) e^{-j2\pi(\frac{ux}{m} + \frac{vy}{n})}, \quad (10)$$

pro $u = 0, 1, 2, \dots, m-1$ a $v = 0, 1, 2, \dots, n-1$, kde $f(x, y)$ jsou hodnoty úrovně šedi obrazového bodu se souřadnicemi (x, y) v obrázku o velikosti $m \times n$. Inverzní Fourierova transformace je tedy definována jako

$$f(m, n) = \frac{1}{m \cdot n} \sum_{u=0}^{m-1} \sum_{v=0}^{n-1} F(u, v) e^{j2\pi(\frac{ux}{m} + \frac{vy}{n})}, \quad (11)$$

kde $u = 0, 1, 2, \dots, m-1$ a $v = 0, 1, 2, \dots, n-1$ jsou frekvenční proměnné. Dvojměrnou DFT je možné vyjádřit v separované formě dvěma jednorozměrnými FT, (každý řádek je nahrazený

svojí 1D FT a potom je vypočítaná 1D FT každého sloupce) [5]. Výsledkem FT reálné funkce je komplexní funkce, jejíž spektrum je definováno jako

$$|F(u, v)| = \sqrt{R^2(u, v) + I^2(u, v)}, \quad (12)$$

kde $R(u, v)$ je reálná složka a $I(u, v)$ je imaginární složka $F(u, v)$.

Výkonové spektrum (nazývané též výkonová spektrální hustota) je definována jako

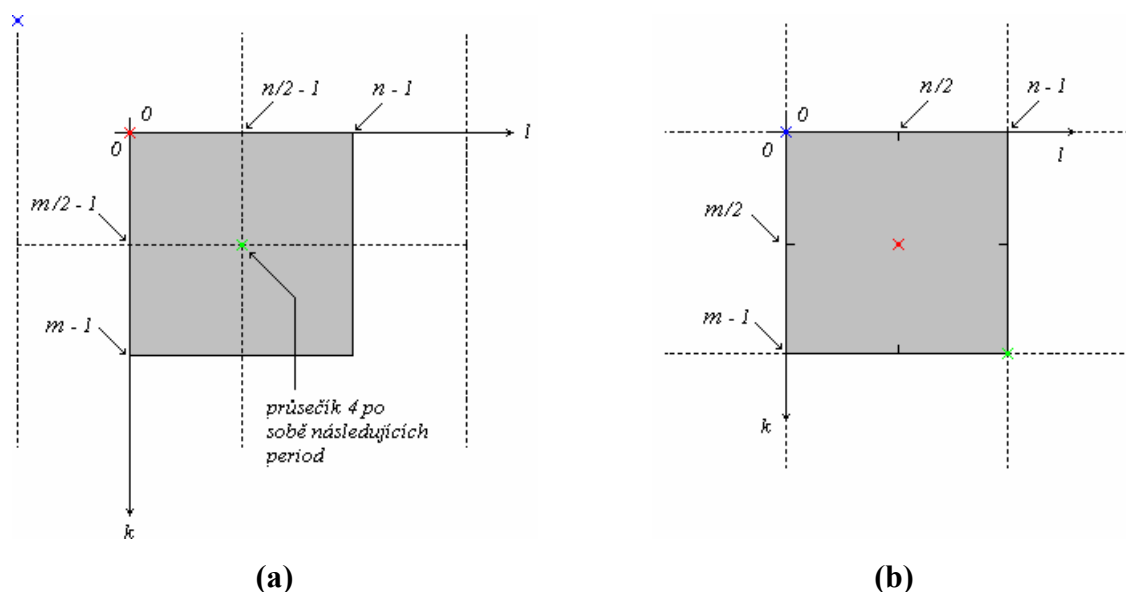
$$P(u, v) = |F(u, v)|^2. \quad (13)$$

Pro zobrazení výkonového spektra jako funkci v úrovních šedi v 8 bitové hloubce je potřebné $P(u, v)$ konvertovat

$$P(u, v) = \log(1 + |F(u, v)|^2). \quad (14)$$

7.2.2 Posunutí

Z definice diskretní Fourierovy transformace vyplývá, že transformace obdélníku bude periodickou funkcí, která bude v rozích nabývat maximálních hodnot. Toto je dáno definicí (10), v níž sumace probíhá od 0 do $n-1$, zatímco „fyzikálně“ by vyhovovala sumace od $-n/2$ do $n/2$, kde by maximální hodnoty nabývaly uprostřed. K zajištění této vlastnosti je možno spočtený obraz posunout o polovinu periody podél obou os (počítačově lze realizovat prohozením jednotlivých čtvrtin) [3], což je naznačeno na obrázku 12.

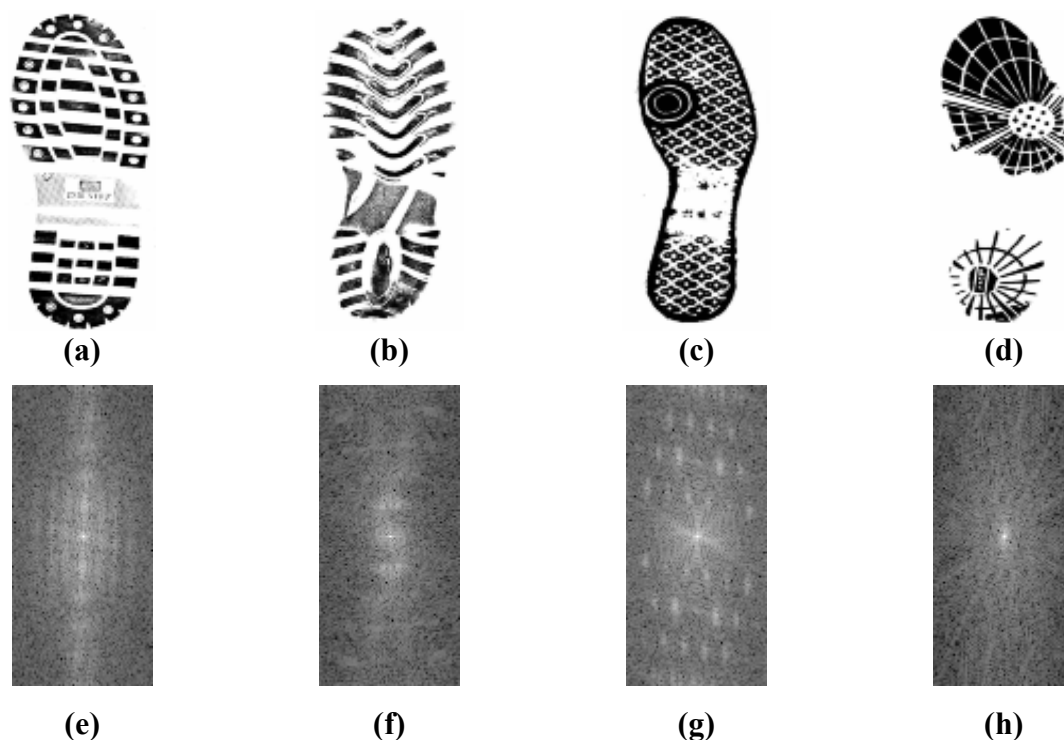


Obrázek 12: Znáznornění posunutí maximálních hodnot spektra DFT

Na obrázku 12(a) jsou hodnoty $F(u,v)$ po DTF jako šedý čtverec. Čárkované čtverce jsou opakující se periody. Šedý čtverec hodnot $F(u,v)$ nyní obklopují 4 po sobě následující periody, které se střetávají v bodě, který je též znázorněn na obrázku 12(a). Na obrázku 12(b) je znázorněno posunutí maximálních hodnot, jak bylo popsáno výše.

7.2.3 Spektrální přístup k vyhledávání v referenční databázi

Fourierova transformace charakterizuje obsah obrazu na základě frekvenčních komponent. Periodicky se opakující příznaky, jako jsou strukturní linie vzorku podešve boty, můžou být pozorované z hodnoty frekvenčních komponent. Jsou lehko rozpoznatelné jako vysoká koncentrace energie frekvenčních komponent v obraze Fourierova frekvenčního spektra. Z toho vyplývá, že každý obrázek z referenční databáze bude mít své charakteristické frekvenční spektrum. Ukázku těchto charakteristických frekvenčních spekter je možno vidět na obrázku 13.



Obrázek 13: Ukázka frekvenčních spekter po DFT obrázků z referenční databáze. (a) Stopa č.1, (b) stopa č.10, (c) stopa č.23, (d) stopa č.71, (e) frekvenční spektrum stopy č.1, (f) frekvenční spektrum stopy č.10, (g) frekvenční spektrum stopy č.23, (h) frekvenční spektrum stopy č.71.

Na základě tohoto faktu lze pomocí korelačního koeficientu mezi frekvenčními spektry obrázků databáze a spektrem obrázku dotazu vyhledávat. Korelační koeficient udává míru podobnosti mezi obrázkem dotazu a obrázkem z referenční databáze.

7.3 Metoda Invariantních momentů

Invariantní momenty v obrazové analýze produkují soubor hlavních vektorů, které jsou invariantní vůči posunutí, natočení a změně velikosti obrazu. Tato metoda má široké využití v extrakci globálního vzhledu pro rozpoznávání obrazů daných jejich velikostí a rozlišením.

Invariantní momenty jsou důležitou metodou pro popis tvaru v počítačovém obrazu. Jsou zde dva typy pro popis obrazu touto metodou:

1. Podle obvodu (kontury) objektu v obraze.
2. Podle obsahu objektu v obraze.

Nejčastěji se při vyhledávání touto metodou využívá vyhledávání podle obrysů [1].

7.3.1 Centrální momenty

Dvou-dimenzionální momenty řádu $(p+q)$, digitálního obrazu o velikosti $m \times n$, digitálního zobrazení $f(x, y)$, kde $(x = 0, 1, 2, \dots, m-1$ a $y = 0, 1, 2, \dots, n-1)$ je definován jako

$$m_{pq} = \sum_{x=0}^{m-1} \sum_{y=0}^{n-1} x^p y^q f(x, y), \quad (15)$$

pro $p, q = 0, 1, 2, \dots$, kde sčítáme přes hodnoty prostorových souřadnic x a y v rozsahu velikosti daného obrazu. Odpovídající *centrální moment* je definován vztahem

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y), \quad (16)$$

kde

$$\bar{x} = \frac{m_{10}}{m_{00}}, \quad \bar{y} = \frac{m_{01}}{m_{00}} \quad (17, 18)$$

Standardizovaný centrální moment řádu $(p + q)$ je definován jako

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma}, \quad (19)$$

pro $p, q = 0, 1, 2, \dots$, a kde

$$\gamma = \frac{p+q}{2} + 1, \quad (20)$$

pro $p + q = 2, 3, \dots$.

Soubor sedmi 2D invariantních momentů, které jsou nezávislé na změně velikosti zrcadlovému přetočení a rotaci obrazu, může být odvozený z těchto rovnic [2]

$$CM1 = \eta_{20} + \eta_{02} \quad (21)$$

$$CM2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (22)$$

$$CM3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (23)$$

$$CM4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (24)$$

$$CM5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 \right] \\ + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right] \quad (25)$$

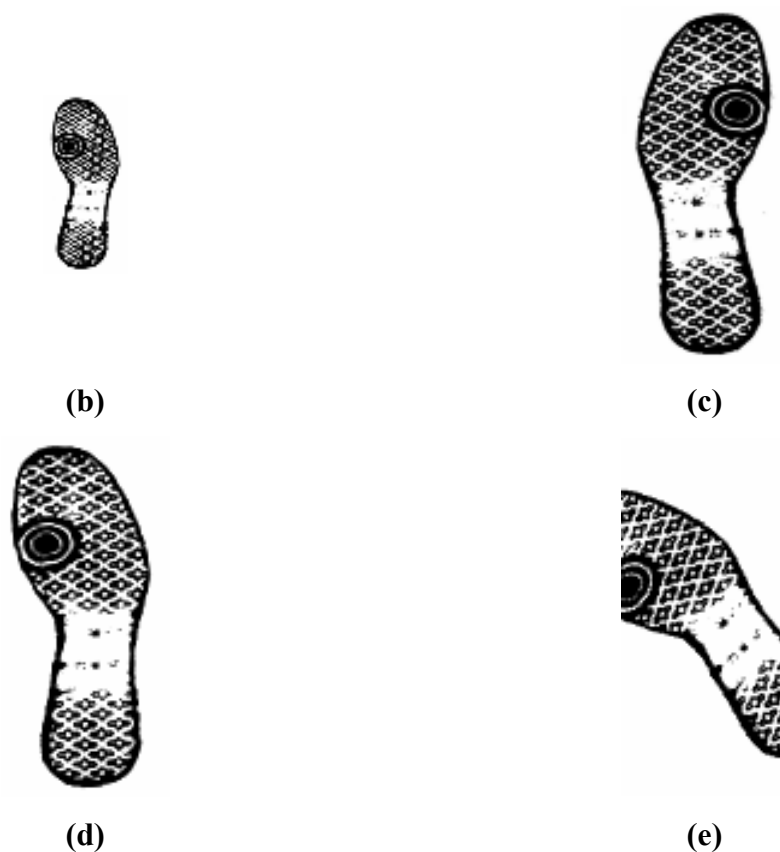
$$CM6 = (\eta_{20} - \eta_{02}) \left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (26)$$

$$CM7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12}) \left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2 \right] \\ + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03}) \left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \right] \quad (27)$$

Na obrázku 14 jsou znázorněny změny obrazu zvolené, stopy z referenční databáze. Hodnoty vypočítaných centrálních momentů pro obrázek 14 jsou zaneseny v tabulce 2.



(a)



Obrázek 14: Stopa č.23 a její úpravy.(a) Originální obraz. (b) Poloviční velikost obrazu. (c) Zrcadlově převrácený obraz. (d) Otočený obraz o 5°. (e) Otočený obraz o 45°.

Tabulka 2: Hodnoty centrálních momentů pro ideální stopu č.23

centrální moment	originální	Změna velikosti na 1/2	zrcadlově převrácený	otočení o 5°	otočení o 45°
CM1	6,6232	6,6289	6,6232	6,6108	6,5636
CM2	14,346	14,36	14,346	14,267	13,881
CM3	24,149	24,303	24,149	23,672	23,073
CM4	23,969	24,127	23,969	23,735	23,199
CM4	48,056	48,373	48,056	47,457	46,337
CM6	31,146	31,311	31,146	30,87	30,177
CM7	54,956	55,103	55,046	55,66	57,047

Z tabulky je zřejmé, že se hodnota centrálních momentů obrázku v závislosti na změně velikosti, zrcadlovém přetočení a otočení se pro danou stopu skoro nezměnila. Z tohoto důvodu se tato metoda jeví jako vhodná k vyhledávání poškozeného obrazu .

7.3.2 Klasifikační stromy

Po spočtení všech sedmi centrálních momentů obrázků databáze, bude zapotřebí podle těchto momentů referenční databázi rozklasifikovat, aby v ní šlo podle spočtených momentů vyhledávat. K tomu budou použity klasifikační stromy.

Klasifikační stromy jsou neparametrické modely klasifikace a organizace dat. Jejich hlavní výhoda je snadná aplikace a efektivita vyhledávání. Jsou dobrým kompromisem mezi srozumitelností a přesností vyhledávání. Vytvářejí se vždy nad již existující datovou strukturou. Klasifikační strom je takové uspořádání prvků (uzlů, nodes), ve kterém lze rozeznat předchůdce (rodiče - parent) a následovníky (děti - children). Každý prvek může mít nejvýše jednoho předchůdce a několik následovníků. Kořenem (root) nazýváme takový prvek, který nemá předchůdce. V každém stromu se nachází jen jeden kořen. Naopak listy (leafs) jsou takové prvky, které nemají žádného následovníka. Má-li strom jen jeden prvek, je tento kořenem i listem zároveň. Klasifikační stromy patří k do skupiny binárních stromů. Binární stromy mají nejvíce dva následovníky. Binární stromy v programování slouží jako struktura pro uchovávání dat. Data se do stromu uchovávají jako jednotlivé uzly tak, že jsou v jistém smyslu seřazeny. Pro každý uzel pak platí, že jeho levý následovník je menší a pravý následovník větší než on sám. Z takového uložení dat vyplývá, že nalézt hledanou položku v binárním stromu bude velmi snadné. Stačí začít strom procházet od kořene [18].

8 EXPERIMENTÁLNÍ ČÁST

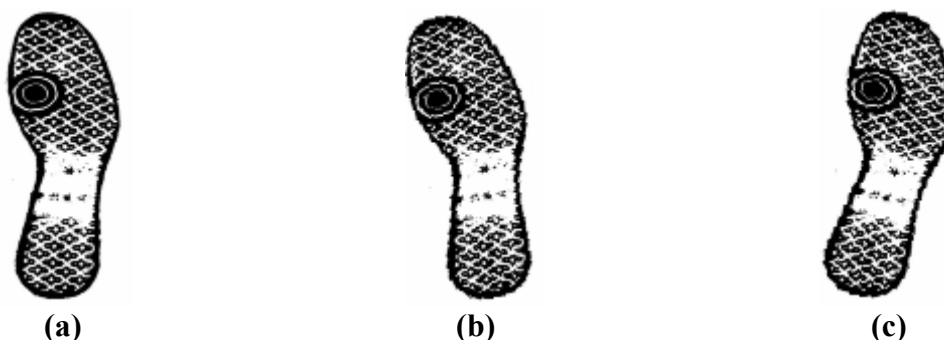
8.1 EXPERIMENT 1 - Použití metody LSI na vybranou skupinu deseti stop

Z databáze obrázků 454 stop bylo vybráno deset stop, které jsou zobrazeny na obrázku 15 a reprezentují různorodost vzorů podešve bot.



Obrázek 15: Ukázka výběru deseti stop, reprezentujících celou databázi. (a) stopa č.1, (b) stopa č.10, (c) stopa č.19, (d) stopa č.23, (e) stopa č.71, (f) stopa č.83, (g) stopa č.109, (h) stopa č.112, (i) stopa č.113, (j) stopa č.114.

První experiment měl určit jak je vyhledávání touto metodou citlivé na případném natočení hledané stopy od svislé osy (obrázky tedy nebyly jinak upravovány, byly pouze natáčeny) tzn., že se zjišťoval interval možného natočení obrázku od svislé osy, aniž by došlo k výraznému zhoršení úspěšnosti vyhledávání. Natočení obrazu je znázorněno na vybrané stopě na obrázku 16.



Obrázek 16: Ukázka intervalu natočení, stopa č.23. (a) Stopa č.23, (b) stopa č.23 natočená o $+10^\circ$ do leva od svislé osy, (c) stopa č.23 natočená o -10° do prava od svislé osy

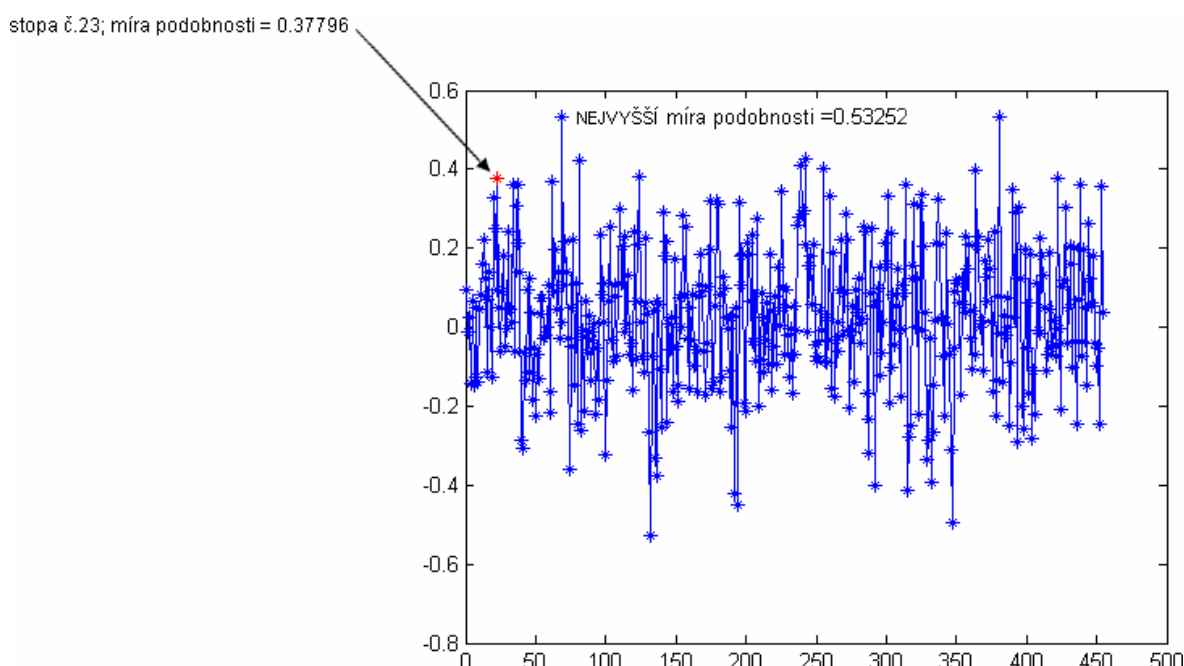
Dalším sledovaným parametrem byla úspěšnost vyhledávání. Ta se zjišťovala následovně:

- 1) Spočetla se míra podobnosti mezi natočeným obrázkem a všemi obrázky v databázi.
- 2) Tyto hodnoty se uložily do sloupcového vektoru (454×1) a byly podle velikosti sestupně seřazeny.
- 3) V takto seřazených hodnotách se hledalo na jaké pozici se nachází hodnota míry podobnosti hledaného obrázku a jeho protějšku z referenční databáze.

Tabulka 3: Příklad výsledků sledovaných parametrů. Stopa č. 23, natočení 10° do leva od svislé osy.

Pozice hledaného obrázku	Index hledaného obrázku	Míra podobnosti hledaného obrázku	Pozice hledaného obrázku	Index hledaného obrázku	Míra podobnosti hledaného obrázku	Pozice hledaného obrázku	Index hledaného obrázku	Míra podobnosti hledaného obrázku
1.	69	0,53252	16.	453	0,35822	31.	428	0,30422
2.	381	0,53115	17.	390	0,34963	32.	395	0,30376
3.	243	0,42715	18.	225	0,34566	33.	110	0,29973
4.	81	0,42116	19.	326	0,3377	34.	241	0,29419
5.	239	0,40876	20.	260	0,33334	35.	392	0,29162
6.	255	0,39988	21.	301	0,33202	36.	141	0,2911
7.	364	0,39642	22.	20	0,32586	37.	271	0,28799
8.	124	0,3815	23.	337	0,32415	38.	240	0,28677
9.	23	0,37796	24.	179	0,32135	39.	155	0,28427
10.	422	0,37722	25.	174	0,31839	40.	238	0,27715
11.	62	0,36792	26.	195	0,3148	41.	208	0,27559
12.	438	0,36254	27.	320	0,31211	42.	444	0,26236
13.	38	0,36252	28.	180	0,31035	43.	237	0,25877
14.	314	0,36191	29.	36	0,30619	44.	284	0,25605
15.	34	0,36096	30.	324	0,30552	45.	157	0,25554

Jako kladný výsledek bylo považováno umístění do 45. místa ze 454 obrázků v databázi. Což odpovídá zhruba 10% původní databáze. V tabulce 3 jsou názorně vidět výsledky vyhledávání. Pro příklad byla k vyhledávání použita stopa č.23, která byla natočena o 10° doleva od svislé osy a zvolené $k = 32$. Hodnoty pro vyhledávaný obrázek jsou v tabulce zvýrazněny. Výsledky byly vynášeny do grafů, kde na ose x byly vyneseny indexy obrázků databáze a na ose y hodnoty míry podobnosti hledaného obrázku s obrázky databáze, jak je možno vidět na obrázku 17.



Obrázek 17: Graf výsledků vyhledávání (vyhledávaná stopa č. 23, natočená o 10° do prava, $k = 32$)

8.1.1 Závěr experimentu 1

V polovině případů se osvědčilo $k \leq 64$ a to pro stopy č.1, 19, 83, 112, 113 – obrázek 15(a), (c), (f), (h), (i). Těmto vzorkům tedy nejlépe vyhovovalo vyhledávání podle hrubých rysů.

Ve druhé polovině případů se osvědčilo $k \geq 128$, pro stopy č.10, 23, 71, 109, 114 – obrázek 15(b), (d), (e), (g), (j). Těmto vzorkům tedy nejlépe vyhovovalo vyhledávání podle jemných detailů.

Pro většinu stop se ukázal vyhovující interval natočení, při které nedošlo k výraznému zhoršení úspěšnosti vyhledávání $\pm 10^\circ$ do svislé osy, ale až po nalezení individuálního k pro

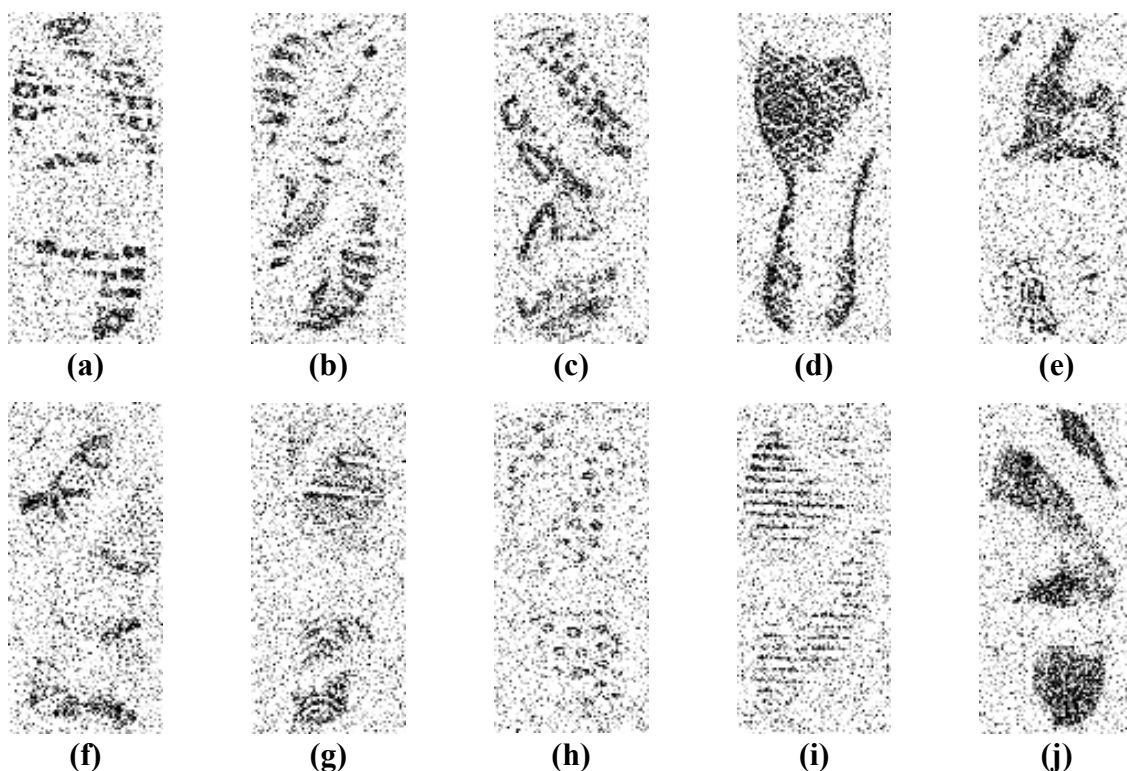
individuální vzorek. Jestliže bychom chtěli použít nějaké univerzální k , pak by se rozpětí tohoto intervalu razantně zúžilo. Doba vyhledávání je uvedena v tabulce 4.

Tabulka 4: Přibližná doba vyhledávání pro použité hodnoty k .

k	čas [s]
48	120
64	190
128	320
256	700

8.2 EXPERIMENT 2 - Použití metody LSI na vybranou skupinu deseti poškozených stop

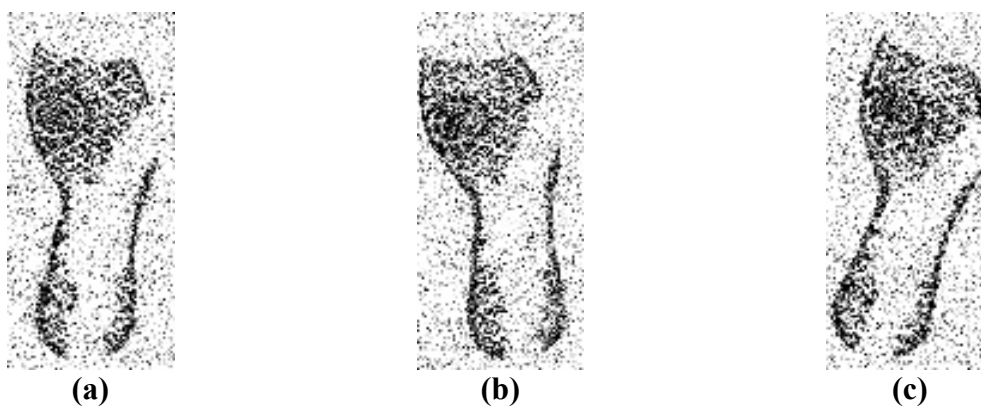
Vybraných deset stop, které jsou zobrazeny na obrázku 15 a reprezentují různorodost vzorů podešve, bylo vyhledáváno v referenční databázi pomocí metody LSI.



Obrázek 18: Ukázka výběru deseti stop, reprezentujících celou databázi, které byly degradovány odmazáním a použitím GBŠ. (a) stopa č.1, (b) stopa č.10, (c) stopa č.19, (d) stopa č.23, (e) stopa č.71, (f) stopa č.83, (g) stopa č.109, (h) stopa č.112, (i) stopa č.113, (j) stopa č.114.

Avšak nyní byla kvalita těchto deseti stop degradována subjektivním odmazáním určitých částí a přidáním GBŠ střední intenzity. Tuto degradaci je možno vidět na obrázku 18.

Takto degradované obrázky pak byly natáčeny na intervalu $\pm 10^\circ$ od svislé osy. Tento interval natočení je vyobrazen na obrázku 19. Krok natočení byl 1° . Pro každý tento krok byly spočítány příslušné míry podobnosti mezi obrázky databáze a obrázkem dotazu.. Dále pro každý krok natočení bylo zjišťováno maximum ze spočítaných měr podobnosti pro celou databázi a k tomuto maximu příslušný index obrázku. Tyto hodnoty byly pro každý hledaný obrázek vyneseny do tabulek. Příklad takové tabulky je možno vidět v tabulce 5. Při vyhledávání v tomto experimentu byla použita hodnota $k = 128$.



Obrázek 19: Ukázka intervalu natočení obrázku (stopa č.23). (a) Natočení 0° , (b) Natočení $+10^\circ$, (c) Natočení -10° .

V tabulce 5 je v prvním sloupci udaná hodnota natočení vůči svislé ose (ve stupních), v druhém sloupci index obrázku (číslo stopy) pro nalezené maximum míry podobnosti a ve třetím sloupci maximum míry podobnosti pro dané natočení. Dále bylo ještě nalezeno maximum míry podobnosti na celém intervalu natočení (maximum z maxim), tj. v tabulce zvýrazněno. Z tabulky 5 je též zřejmé, že v tomto případě (vyhledávání poškozené stopy č.23) bylo vyhledávání úspěšné, protože nejvyšší hodnoty maxim měr podobnosti odpovídá indexu obrázku stopy č.23.

Tabulka 5: Tabulka pro maxima míry podobnosti k příslušnému natočení hledaného obrázku (stopa č.23)

natočení [°]	index obrázku	míra podobnosti
-10	205	0,39255
-9	198	0,38975
-8	453	0,41414
-7	453	0,3713
-6	239	0,39702
-5	205	0,41646
-4	23	0,42295
-3	23	0,48031
-2	23	0,57243
-1	23	0,64288
0	23	0,67504
1	23	0,63007
2	23	0,51754
3	23	0,43725
4	441	0,44137
5	447	0,4208
6	55	0,40784
7	55	0,4456
8	55	0,41406
9	55	0,44192
10	55	0,46658

8.2.1 Závěr experimentu 2

Vyhledávání proběhlo úspěšně v případě, kdy maximum míry podobnosti na celém intervalu natočení (maximum z maxim) odpovídalo indexu vyhledávané stopy. To se při použití hodnoty $k = 128$ podařilo v 9 z 10 případů. U vyhledávání poškozené stopy č.109 musela být hodnota nastavena na $k = 256$, pak i tato stopa byla vyhledána úspěšně.

8.3 EXPERIMENT 3 - Použití metody LSI na reálnou stopu

První reálnou stopu, hledanou v databázi stop pomocí metody LSI byla zvolena stopa označenou jako 1364, která již dříve byla určena a odpovídá stopě, v referenční databázi označené jako stopa č. 439, zobrazeno na obrázku 20(a),(b).



Obrázek 20: Porovnání reálného a ideálního otisku boty. (a) Předupravený otisk boty. (b) Ideální otisk boty z databáze stopy č. 439.

Tato stopa byla následovně pro zlepšení vyhledávání subjektivně v grafickém programu upravena, aby co nejvíce odpovídala ideálním otiskům bot z databáze a to převrácením stopy o 90° doleva, zrcadlovým přetočením, zvýšením kontrastu a jasu, odmazáním přebytečných černých ploch a vycentrování svislé osy stopy. Tyto úpravy lze vidět na obrázku 21. Potom byla resamplována velikost na formát obrázků databáze. Nakonec byla hledaná stopa ještě rozdělena pro potřeby do všech podskupin databáze, což je znázorněno na obrázku 22.



(a)



(b)



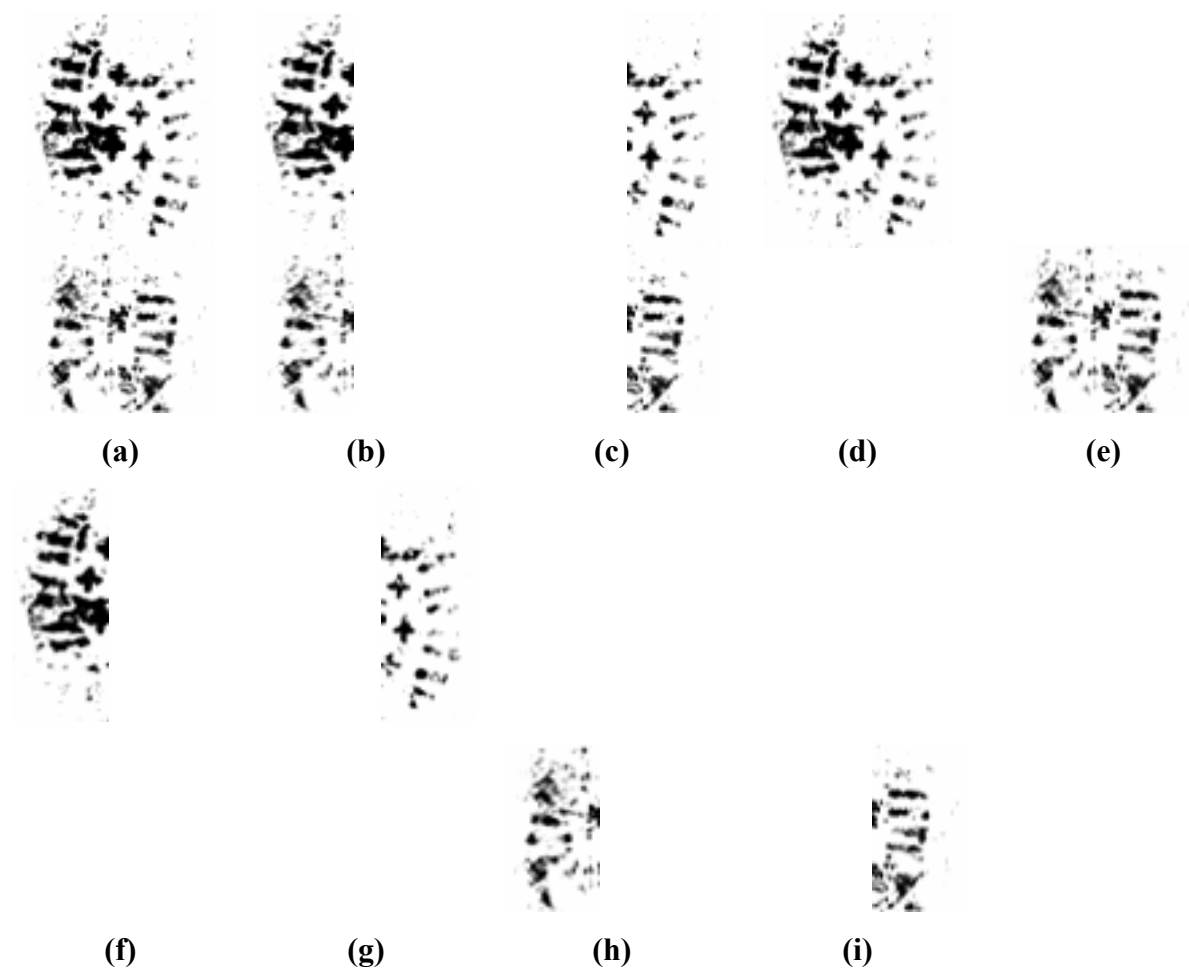
(c)



(d)



Obrázek 21: Předzpracování reálné stopy před samotným vyhledáváním. (a) Originální reálná stopa. (b) Převrácení o 90° do leva. (c) Zrcadlové převrácení. (d) Úprava kontrastu a jasů. (e) Odmazání (subjektivní) černých ploch a vycentrování svislé osy stopy



Obrázek 22: Názorná ukázka podskupin databáze pro stopu č. 23. (a) Full – celá, (b) HalfLeft - levá polovina, (c) HalfRight - pravá polovina, (d) HalfTop - horní polovina, (e) HalfBottom - dolní polovina, (f) LeftTop - levá horní čtvrtina, (g) RightTop - pravá horní čtvrtina, (h) LeftBottom - levá dolní čtvrtina, (i) RightBottom - pravá dolní čtvrtina.

Tato stopa byla natáčena na intervalu $\pm 10^\circ$ od svislé osy. Krok natočení byl 1° a pro každý tento krok byly spočítány příslušné míry podobnosti, které byly sestupně seřazeny podle velikosti do vektoru. V takto seřazeném vektoru se hledalo, na jakém místě se umístily míry podobnosti pro stopu č. 439. Výsledky vyhledávání této stopy je možno vidět v tabulce 6.

Druhou reálnou stopou, hledanou v databázi stop pomocí metody LSI byla zvolena stopa označená jako 1543. Tato stopa byla dříve rovněž určena. Avšak její určení nebylo úplně jednoznačné, ale odpovídá třem stopám v referenční databázi označených jako stopa č.: 3, 7 a 270, možno vidět na obrázku 23. I tato stopa byla opět natáčena na intervalu $\pm 10^\circ$ od svislé osy. Krok natočení byl 1° a pro každý krok byly spočítány příslušné míry podobnosti, které byly sestupně seřazeny podle velikosti do vektoru. V takto seřazeném vektoru se hledala pozice míry podobnosti pro stopy č. 3, 7, 270. Tyto hodnoty se pak vynesly do tabulek, jak je ukázáno v tabulce 7 a to pro všech 9 podskupin databáze. Při vyhledávání obou reálných stop byla použita hodnota $k = 128$.



Obrázek 23: Vyobrazení reálného otisku boty a možných, odpovídajících, ideálních otisků bot. (a) Předzpracovaný otisk sejmутý na místě činu. (b) Ideální otisk z databáze stopy č. 3. (c) Ideální otisk z databáze stopy č. 7. (d) Ideální otisk z databáze stopy č. 270.

Tabulka 6: Výsledky vyhledávání reálné stopy odpovídající ideální stopě č 439.

úhel natočení [°]	Full	Half Bottom	HalfTop	HalfLeft	Half Right	Left Bottom	Right Bottom	LeftTop	RightTop
-10	343.	366.	395.	175.	424.	77.	295.	1.	190.
-9	364.	379	399.	143.	422.	111.	286.	1.	185.
-8	386.	363.	406.	103.	408.	88.	243.	2.	218.
-7	407.	343.	411.	96.	390.	127.	265.	3.	145.
-6	419.	308.	397.	88.	338.	130.	234.	7.	113.
-5	422.	249.	407.	71.	223.	111.	173.	8.	73.
-4	432.	220.	371.	63.	153.	166.	154.	8.	37.
-3	392.	193.	286.	76.	66.	163.	110.	8.	27.
-2	386.	90.	184.	56.	71.	163.	109.	12.	20.
-1	332.	64.	186.	62.	63.	170.	65.	20.	33.
0	351.	70.	189.	107.	61.	170.	79.	19.	27.
1	361.	72.	201.	195.	66.	172.	87.	19.	31.
2	344.	47.	204.	209.	188.	169.	56.	26.	38.
3	311.	29.	272.	265.	240.	167.	75.	44.	59.
4	324.	8.	257.	281.	334.	195.	101.	76.	65.
5	293.	6.	234.	218.	419.	246.	115.	153.	81.
6	348.	4.	256.	182.	442.	234.	98.	225.	65.
7	350.	1.	276.	143.	452.	240.	84.	334.	97.
8	378.	1.	255.	203.	454.	270.	84.	329.	70.
9	348.	1.	288.	182.	454.	267.	74.	353.	57.
10	360.	4.	328.	223.	454.	331.	56.	359.	50.

V tabulce 6 a 7 bylo, pro lepší orientaci, zvýrazněno nulové natočení (natočení o 0°).

Tabulka 7: Ukázka tabulky hodnot umístění hledaných stop v závislosti na natočení (podskupina Full).

úhel natočení [°]	stopa č.3	stopa č.7	stopa č.270
-10	289.	222.	188.
-9	273.	182.	177.
-8	221.	148.	156.
-7	153.	101.	117.
-6	105.	108.	92.
-5	79.	91.	102.
-4	50.	70.	87.
-3	35.	67.	108.
-2	20.	96.	71.
-1	18.	91.	42.
0	13.	135.	30.
1	13.	145.	27.
2	9.	155.	12.
3	8.	179.	6.
4	8.	185.	7.
5	6.	267.	5.
6	5.	307.	3.
7	4.	352.	9.
8	7.	353.	29.
9	17.	346.	75.
10	20.	308.	152.

Pro hrubý přehled úspěšnosti vyhledávání ve všech 9-ti podskupinách databáze byla zhotovena tabulka 8, kde byla úspěšnost vyjádřena níže uvedenou, jednoduchou symbolikou.

Tabulka 8: Výsledky úspěšnosti vyhledávání reálné stopy 1543 pro všechny podskupiny databáze.

stopa č.	Full	Half Bottom	HalfTop	HalfLeft	Half Right	Left Bottom	Right Bottom	LeftTop	RightTop
3	••••		••••	••••	•••	••••		••••	••
7		•••	•••		•••	••••	••••	•••	•••
270	••••	••••	•	••••		••••	•••		

•••• Umístění vyhledávané stopy na 1.- 10. místě.

••• Umístění vyhledávané stopy na 11.- 20. místě.

•• Umístění vyhledávané stopy na 21.- 30. místě.

• Umístění vyhledávané stopy na 31.- 45. místě.

prázdné políčko Umístění vyhledávané stopy na >45. místě.

8.3.1 Závěr experimentu 3

Jak je vidět v tabulce 8, se ve většině případů vyhledávané stopy umístily ve vektoru měr podobností seřazeném sestupně podle velikosti do 10. místa. Tzn., že vyhledávání druhé reálné stopy je uspokojivé. Naopak vyhledávání první reálné stopy proběhlo neuspokojivě. Umístění hledané stopy bylo ve většině případů > 45. Z toho vyplývá, že úspěšnost vyhledávání reálných stop touto metodou, bude velice záviset na kvalitě sejmuté stopy.

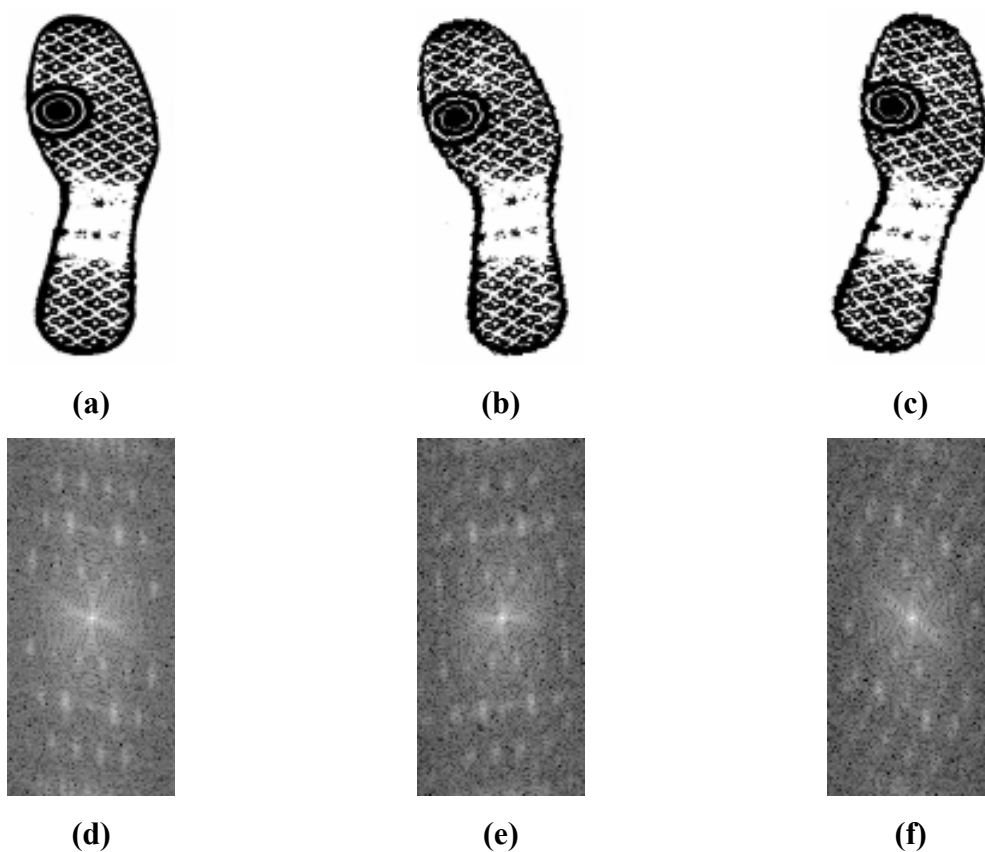
8.4 EXPERIMENT 4 - Použití metody DFT na vybranou skupinu deseti stop

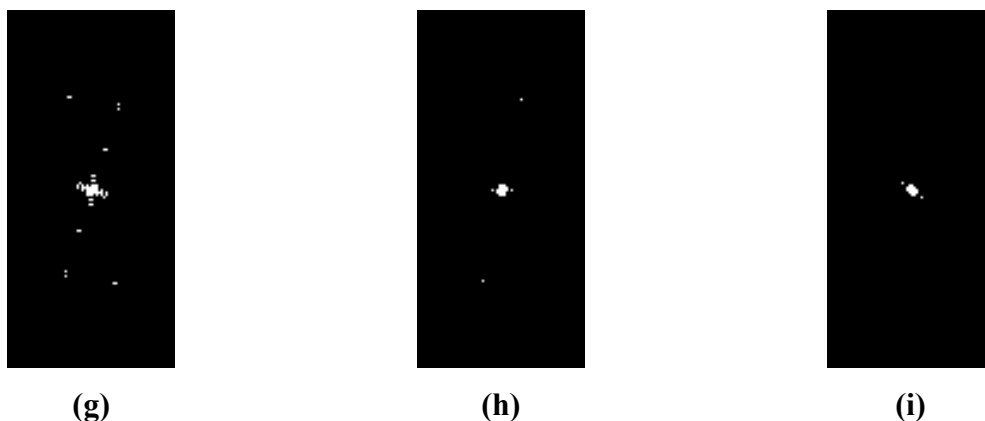
Opět byla pro srovnání účinnosti a spolehlivosti zkoušených metod vyhledávání použita stejná skupina deseti stop, jako v předchozím případě při vyhledávání pomocí LSI. A vyhledávaný obraz byl opět natáčen na intervalu $\pm 10^\circ$ od svislé osy, zobrazeno na obrázku 16. První experiment s touto metodou měl, jako v experimentu 1 určit, jak je vyhledávání touto metodou citlivé na natočení vyhledávaného obrázku. Tento parametr se zjišťoval následovně :

- 1) Spočetlo se spektrum DFT natočeného obrázku dotazu a všech obrázků v databáze.
- 2) Dále se provedla korelace mezi spektrem obrázku dotazu a všech obrázků v databáze.

- 3) Tyto hodnoty se uložily do sloupcového vektoru (454×1) a byly podle velikosti sestupně seřazeny.
- 4) V takto seřazených hodnotách se hledalo, na jaké pozici se nachází hodnota korelace hledaného obrázku a jeho protějšku z referenční databáze.

Dalším sledovaným parametrem byla závislost úspěšnosti vyhledávání na oprahování spekter stop po DFT, tzn., že se výše popsany postup prováděl nejdříve na neoprahovaných a potom na oprahovaných spektrech, kde byl použitý práh = 0,7. Nakonec se výsledky vzájemně porovnály. Změna spekter je vyobrazena na obrázku 24. Ukázku výsledků tohoto experimentu je možno vidět v tabulce 9. Maximální hodnoty korelací při vyhledávání jsou v tabulce zvýrazněny.





Obrázek 24: Ukázka změny spektra obrázku po DFT v závislosti na natočení. (a) Stopa č.23, (b) stopa č.23 natočená o $+10^\circ$ do leva, (c) stopa č.23 natočená o -10° do prava (d) spektrum po DFT k obrázku 24(a), (e) spektrum po DFT k obrázku 24(b), (f) spektrum po DFT k obrázku 24(c), (g) oprahované spektrum po DFT k obrázku 24(a), (h) oprahované spektrum po DFT k obrázku 24(b), (i) oprahované spektrum po DFT k obrázku 24(c).

Tabulka 9: Maximální hodnoty korelace obrázků spekter po DFT hledaného obrázku s obrázky databáze v závislosti na natočení (vyhledávaná stopa č.23).

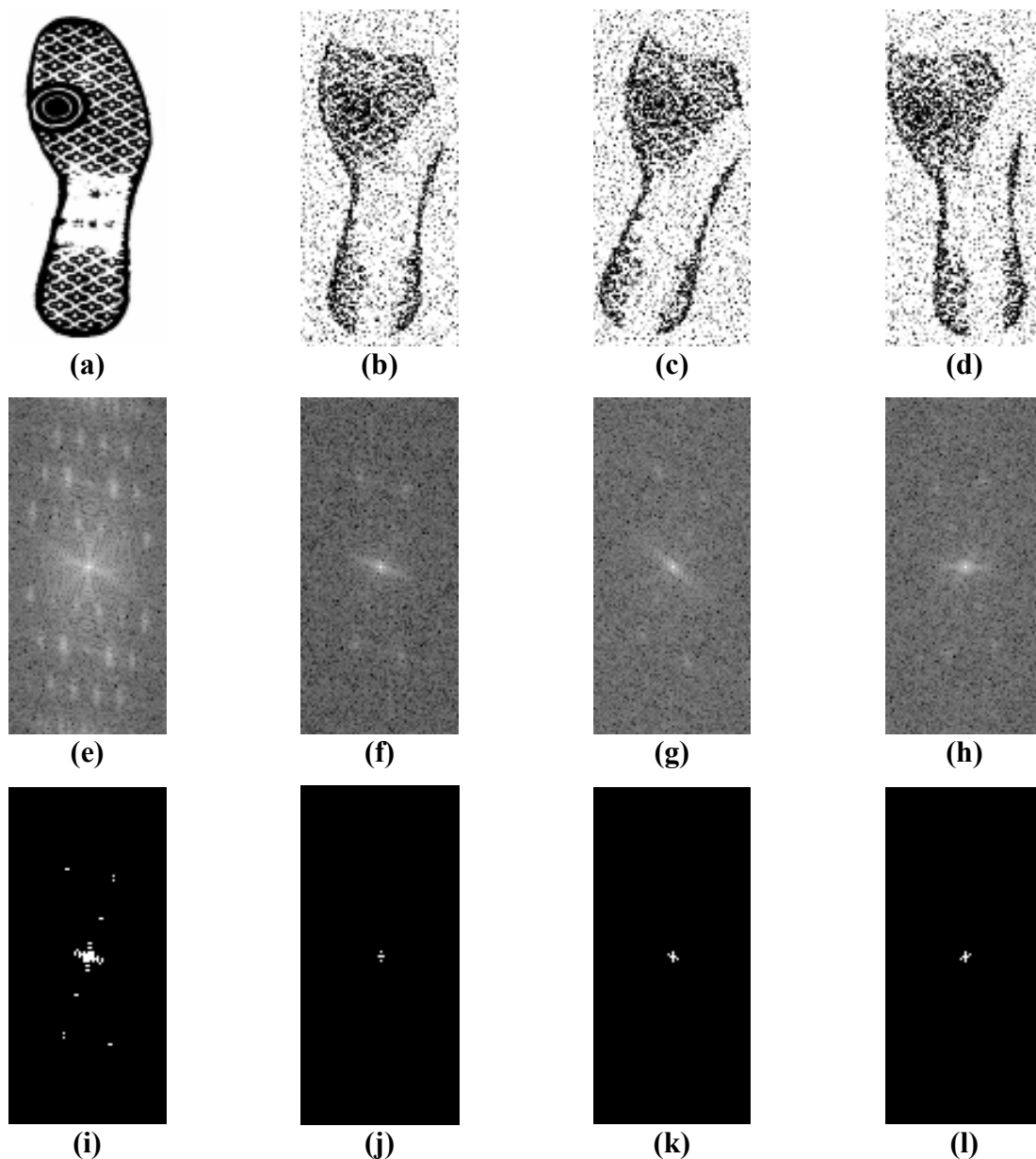
natočení [°]	neoprahovaná spektra po DFT		oprahovaná spektra po DFT	
	index obrázku	míra podobnosti	index obrázku	míra podobnosti
-10	69	0,30444	93	0,50141
-9	69	0,32246	279	0,47329
-8	69	0,32807	214	0,54232
-7	69	0,34759	214	0,59011
-6	69	0,34606	202	0,49985
-5	239	0,34952	223	0,52095
-4	23	0,36797	223	0,54529
-3	23	0,42124	214	0,51416
-2	23	0,50573	223	0,50922
-1	23	0,61078	23	0,56071
0	23	0,69832	23	0,68316
1	23	0,62324	23	0,60068
2	23	0,49469	296	0,54232
3	23	0,42425	57	0,515
4	23	0,37482	57	0,52752
5	144	0,37245	253	0,54912
6	144	0,36428	253	0,53364
7	144	0,35361	253	0,50365
8	144	0,34977	311	0,51585
9	137	0,33006	101	0,50775
10	144	0,32572	152	0,55108

8.4.1 Závěr experimentu 4

Vyhledávání proběhlo se 100% úspěšností při vyhledávání za použití korelace neoprahaných spekter DFT. Nejvyšší hodnota korelace je vždy mezi degradovaným obrázkem a jeho originálem z referenční databáze. Při vyhledávání za použití korelace oprahaných spekter DFT vyhledávání nedopadlo pozitivně. Většinou nebyla při příslušném natočení hodnota korelace mezi degradovaným obrázkem a jeho originálem v databázi maximální, tzn., že se nenacházela na prvním místě.

8.5 EXPERIMENT 5 - Použití metody DFT na vybranou skupinu deseti poškozených stop

Vyhledávání probíhalo stejným způsobem jako v předchozím experimentu 4, ale vyhledávaný obrázek byl poškozen odmazáním určitých ploch a přidáním GBŠ. Na obrázku 25(a) – (d) je znázorněna ukázka obrázku stopy č.23 a degradace jeho kvality. Obrázek 25(e) - (h) zobrazuje spektrum DFT pro použitý stupeň degradace obrázku. Na obrázku 25(i) – (l) na vidět příslušná oprahaná spektra DFT. V tabulce 10 jsou pro ukázku znázorněny výsledky vyhledávání poškozené stopy č.23. Pro porovnání jsou výsledky vyhledávání zaneseny jak pro neoprahaná spektra DFT, tak i pro oprahaná spektra DFT. Maximální hodnoty korelací při vyhledávání jsou v tabulce zvýrazněny.



Obrázek 25: Ukázka změny spektra obrázku po DFT v závislosti na degradaci. (a) Stopa č.23, (b) degradovaná stopa č.23, (c) degradovaná stopa č.23 natočená o -10° do prava (d) degradovaná stopa č.23 natočená o $+10^\circ$ do leva, (e) spektrum po DFT k obrázku 25(a), (f) spektrum po DFT k obrázku 25(b), (g) spektrum po DFT k obrázku 25(c), (h) spektrum po DFT k obrázku 25(d), (i) oprahované spektrum po DFT k obrázku 25(a), (j) oprahované spektrum po DFT k obrázku 25(b), (k) oprahované spektrum po DFT k obrázku 25(c), (l) oprahované spektrum po DFT k obrázku 25(d).

Tabulka 10: Maximální hodnoty korelace obrázků spekter po DFT hledaného obrázku s obrázky databáze v závislosti na natočení (vyhledávaná stopa č.23).

natočení [°]	neoprahaná spektra po DFT		oprahaná spektra po DFT	
	index obrázku	míra podobnosti	index obrázku	míra podobnosti
-10	205	0,39255	93	0,46074
-9	198	0,38975	214	0,54164
-8	453	0,41414	418	0,46435
-7	453	0,3713	214	0,4742
-6	239	0,39702	223	0,52894
-5	205	0,41646	223	0,47576
-4	23	0,42295	223	0,54473
-3	23	0,48031	434	0,47418
-2	23	0,57243	23	0,52488
-1	23	0,64288	23	0,67057
0	23	0,67504	23	0,6642
1	23	0,63007	23	0,68329
2	23	0,51754	57	0,55402
3	23	0,43725	93	0,55168
4	441	0,44137	296	0,55266
5	447	0,4208	253	0,52308
6	55	0,40784	296	0,55841
7	55	0,4456	137	0,5615
8	55	0,41406	250	0,52194
9	55	0,44192	137	0,51277
10	55	0,46658	137	0,54553

8.5.1 Závěr experimentu 5

Vyhledávání dopadlo podobně jako v experimentu 4. Se 100% úspěšností při vyhledávání za použití korelace neoprahaných spekter DFT. Nejvyšší hodnota korelace je vždy mezi degradovaným obrázkem a jeho originálem v databázi.

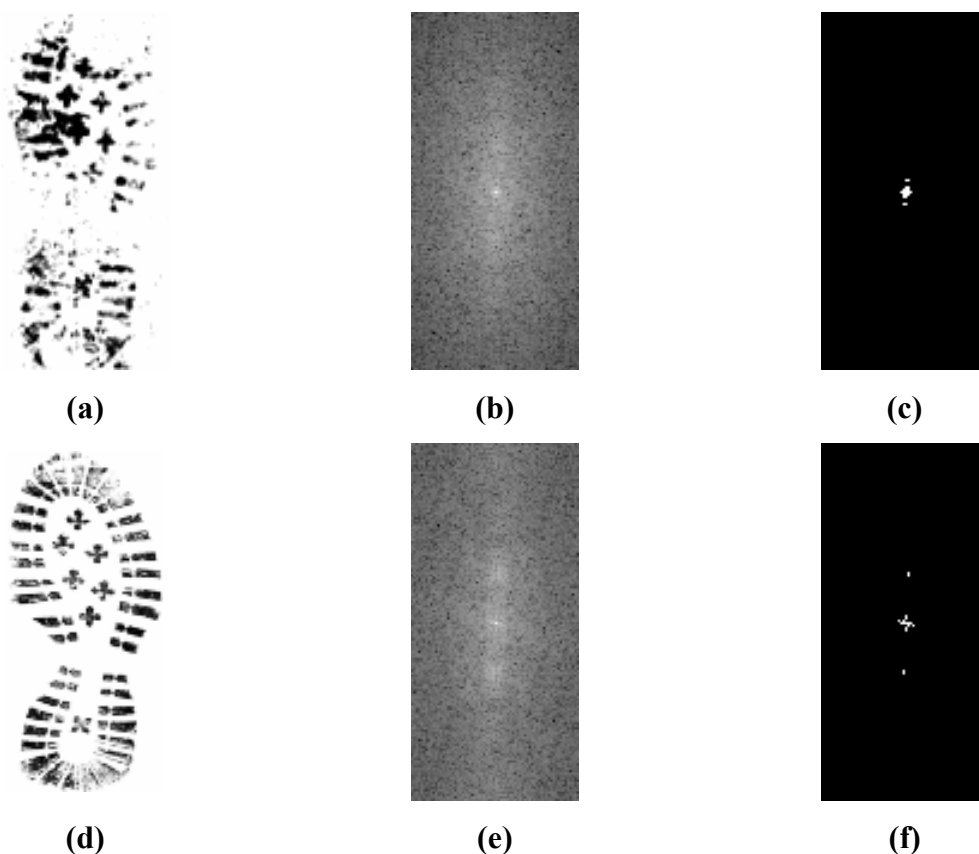
Při vyhledávání za použití korelace oprahaných spekter DFT vyhledávání dopadlo opět o poznání hůře než v případě neoprahaných spekter. Většinou nebyla při příslušném natočení hodnota korelace mezi degradovaným obrázkem a jeho originálem v databázi maximální, tzn. že se nenacházela na prvním místě.

8.6 EXPERIMENT 6 - Použití metody DFT pro vyhledávání realné stopy

Jako první reálnou stopu, hledanou v databázi stop pomocí metody DFT byla jako v experimentu 3 zvolena stopa označena jako 1364, která již dříve byla určena a odpovídá stopě, v naší databázi označené jako stopa č. 439, zobrazeno na obrázku 20(a),(b). Tato stopa

byla následně pro zlepšení vyhledávání upravena, aby co nejvíce odpovídala reálným otiskům bot.

Pro takto předzpracovanou stopu se sestrojilo spektrum DFT a následně byla spočítána korelace mezi spektrem této vyhledávané stopy a spektry obrázků stop databáze. To bylo provedeno ve všech 9-ti podskupinách databáze. Korelace byla pro porovnání s předchozím experimentem provedena pro neoprahaná i oprahovaná spektra. Opět byl při prahování použit práh = 0,7



Obrázek 26: Ukázka porovnání spekter DFT reálné a ideální stopy. (a) Reálná stopa, (b) neoprahané spektrum reálné stopy, (c) oprahované spektrum reálné stopy, (d) ideální stopa, (e) neoprahané spektrum ideální stopy, (f) oprahované spektrum ideální stopy.

Na obrázku 26 je možno mezi sebou vizuálně porovnat spektra reálné a ideální stopy, kde je už na první pohled patrné, že nejsou zcela identická. Výsledky vyhledávání pro tuto stopu jsou zaneseny v tabulce 11 pro neoprahaná spektra a v tabulce 12 pro oprahovaná spektra. V obou tabulkách jsou též uvedeny nejvyšší hodnoty korelací a číslo odpovídající stopy.

Tabulka 11: Výsledky vyhledávání reálné stopy odpovídající ideální stopě č.439 pro neoprahaná spektra.

	Full	HalfBottom	HalfLeft
umístění	362	127	387
hodnota korelace	0,1406	0,2256	0,14381
top korelace / stopa č.	0,33107 / 75	0,32971 / 306	0,41595 / 62
	HalfRight	HalfTop	LeftBottom
umístění	338	308	173
hodnota korelace	0,11891	0,15131	0,17599
top korelace / stopa č.	0,29739 / 400	0,38838 / 205	0,35894 / 240
	LeftTop	RightBottom	RightTop
umístění	359	150	171
hodnota korelace	0,15667	0,28413	0,13531
top korelace / stopa č.	0,48347 / 62	0,43931 / 330	0,28207 / 102

Tabulka 12: Výsledky vyhledávání reálné stopy odpovídající ideální stopě č.439 pro oprahovaná spektra.

	Full	HalfBottom	HalfLeft
umístění	176	32	30
hodnota korelace	0,3297	0,46021	0,44779
top korelace / stopa č.	0,50822 / 342	0,55833 / 162	0,51717 / 160
	HalfRight	HalfTop	LeftBottom
umístění	27	8	276
hodnota korelace	0,3676	0,46004	0,34512
top korelace / stopa č.	0,47354 / 142	0,50058 / 238	0,60312 / 329
	LeftTop	RightBottom	RightTop
umístění	44	159	78
hodnota korelace	0,50805	0,40281	0,35566
top korelace / stopa č.	0,64179 / 314	0,71364 / 412	0,48761 / 408

Druhou reálnou stopu, hledanou v databázi stop pomocí metody DFT, byla opět zvolena stopa označená jako 1543, která byla dříve rovněž určena a odpovídá třem stopám v naší databázi označené jako stopa č.: 3, 7 a 270, možno spatřit na obrázku 23. Do tabulky 13 byly pro ukázkou zaneseny výsledky vyhledávání v podskupině *Full*.

Tabulka 13: Ukázka výsledků vyhledávání reálné stopy odpovídající ideální stopě č.3, 7, 270 pro podskupinu Full.

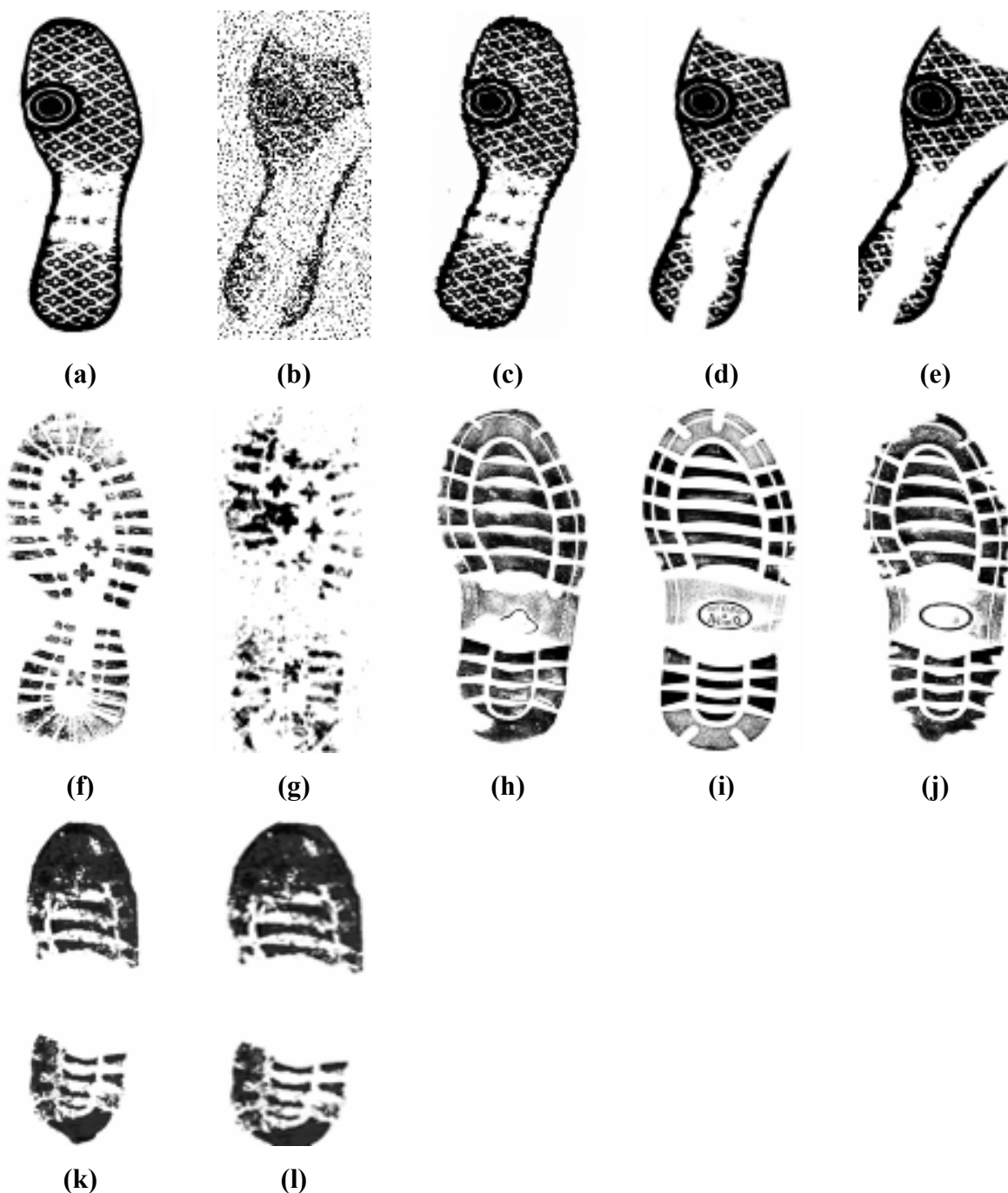
neoprahané spektra			
	stopa č.3	stopa č.7	stopa č. 270
umístění	45	313	86
hodnota korelace	0,41111	0,24588	0,37047
top korelace : 0,50898 – stopa č. 21			
oprahané spektra			
	stopa č.3	stopa č.7	stopa č. 270
umístění	6	159	14
hodnota korelace	0,56913	0,4368	0,538
top korelace : 0,56667 – stopa č. 394			

8.6.1 Závěr experimentu 6

Jak je patrné z tabulek 11-13 je narozdíl od experimentu 4 a 5 vhodnější použít k vyhledávání oprahaná spektra. Avšak vyhledávání už zdaleka není tak uspokojivé, jako v případě experimentu 4 a 5. Zhruba v polovině případů se vyhledávaný obrázek nacházel na místě > 45.

8.7 EXPERIMENT 7 - Použití metody centrálních momentů a klasifikačního stromu

Nejdříve se spočetlo všech sedm centrálních momentů pro každou stopu z referenční databáze tzn., že se vytvořila matice hodnot centrálních momentů o velikosti 454×7 , kde každý řádek této matice představoval vektor sedmi centrálních momentů dané stopy z referenční databáze. Ještě před tím, než byla referenční databáze rozklasifikována pomocí klasifikačního stromu, přidaly se do ní poškozené obrázky stopy č.23 a dvě subjektivně předzpracované reálné stopy, které byly vyhledávány i v experimentech 3 a 6. Tyto stopy byly očíslovány tak, aby pokračovaly v číslování referenční databáze (tedy 455 až 461). Toto je znázorněno na obrázku 27. I pro tyto stopy se spočetly hodnoty centrálních momentů a rovněž byly přidány do matice hodnot centrálních momentů referenční databáze. Výsledná, rozšířená matice tedy byla velikosti 461×7 .

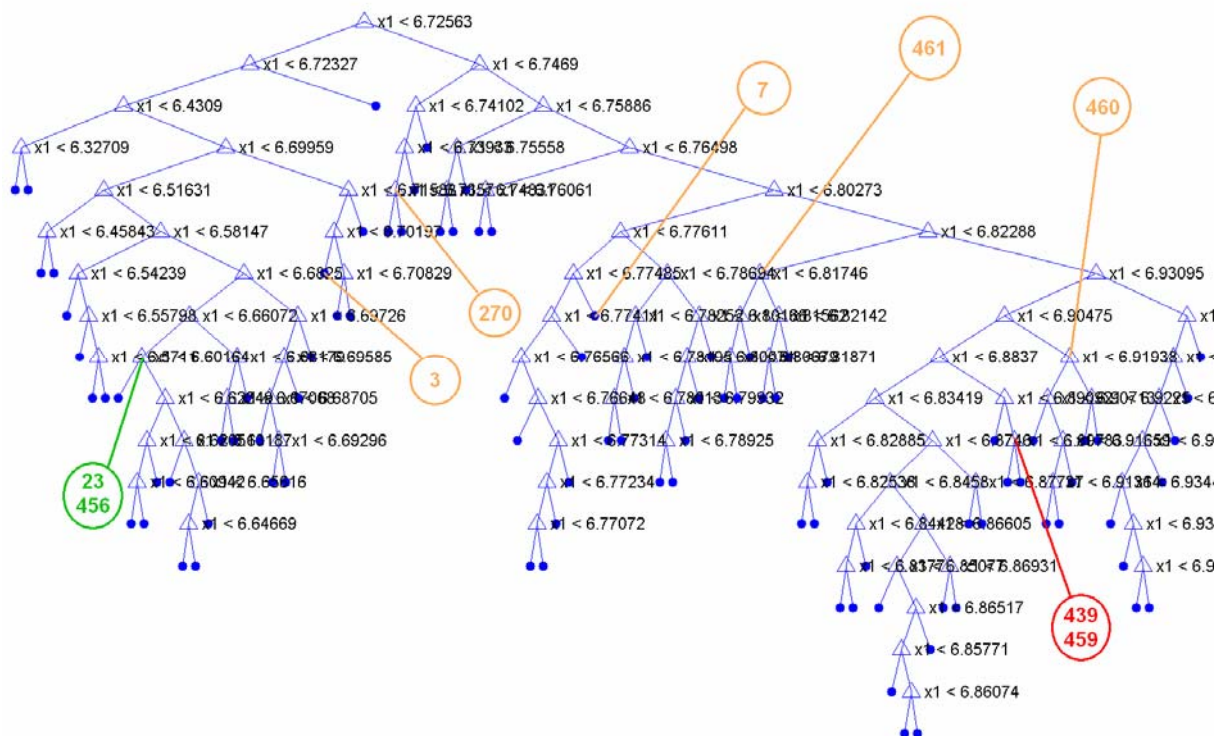


Obrázek 27: Zobrazení vyhledávaných stop v klasifikačním stromu. (a) Stopa č.23, (b) stopa č.455 – subjektivně obmazaná stopa č.23, natočená o 10° doprava + přidán GBŠ, (c) stopa č.456 - stopa č.23, natočená o 10° doprava, (d) stopa č.457 - subjektivně obmazaná stopa č.23, natočená o 10° doprava, (e) stopa č.458 - subjektivně obmazaná stopa č.23, natočená o 20° doprava, (f) stopa č. 439, (g) stopa č.459 – reálná stopa odpovídající stopě č.439 z referenční databáze, (h) stopa č.3, (i) stopa č.7, (j) stopa č.270, (k) stopa č.460 – reálná stopa odpovídající stopě č.3, 7 a 270 z referenční databáze, (l) stopa č.461 – stopa č.460 se zvětšenou šířkou.

Hodnoty centrálních momentů z výchozí matice CM pak sloužily jako rozdělovací podmínky pro klasifikační strom. Funkci vyvolání klasifikačního stromu v MATLABu vykonává příkaz $T = \text{treefit}(X,y)$, kde je v tomto případě X matice hodnot centrálních

momentů a y jsou indexy stop databáze. V takto sestrojeném klasifikačním stromu se zjišťovalo, jestli se vyhledávaný obrázek nachází ve stejné větvi stromu jako obrázek z referenční databáze odpovídající hledanému obrázku. Na obrázku 28 je vidět výsledný klasifikační strom.

CMall



Obrázek 28: Klasifikační strom pro matice hodnot centrálních momentů.

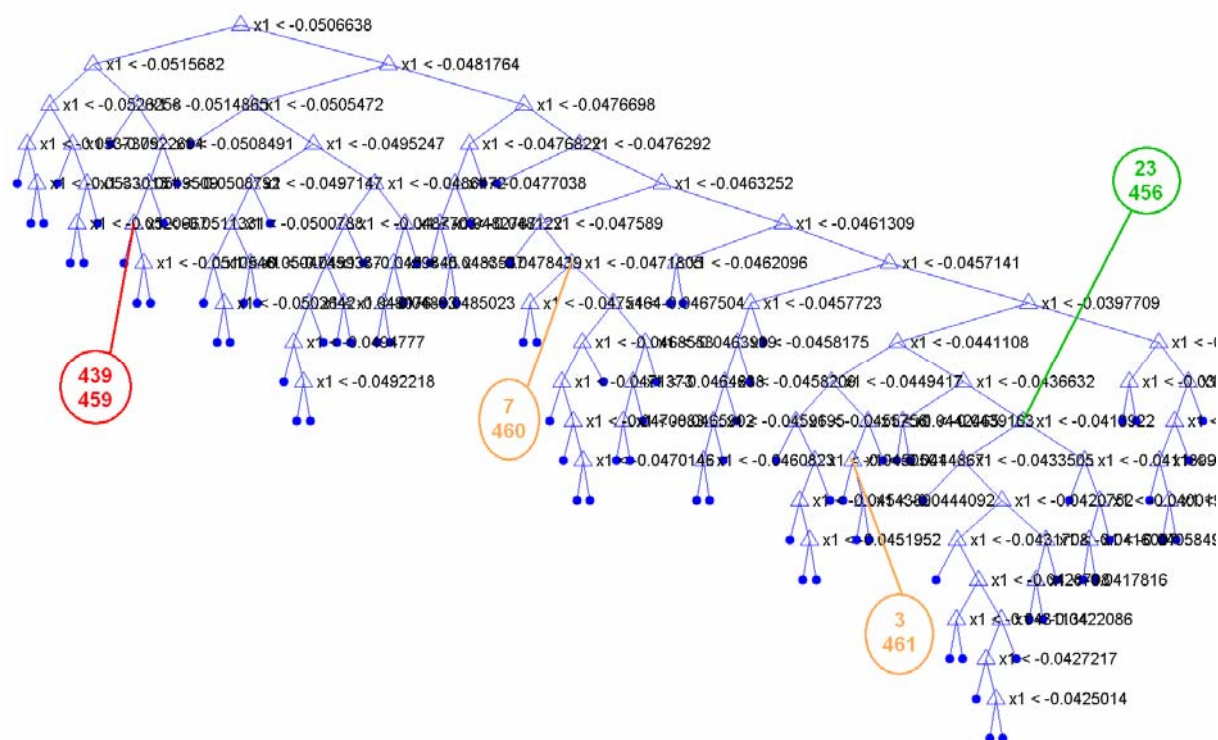
V klasifikačním stromu na obrázku 28 je naznačena oblast, kde se nachází stopy č. 23, 455 – 458 zelenou popiskou, stopy č. 439, 459 červenou popiskou a stopy č. 3, 7, 270, 460 a 461 oranžovou popiskou.

8.7.1 Závěr experimentu 7

Takto zkonstruovaný klasifikační strom použil jen hodnoty CM1. Taková klasifikace pro tento případ nevyhovující. Klasifikace pomocí klasifikačního stromu a matice hodnot centrálních momentů databáze je tedy nepoužitelná.

8.8 EXPERIMENT 8 - Použití SVD rozkladu a klasifikačního stromu pro vyhledávání v referenční databázi

Postup byl stejný jako u experimentu 7, jen se při konstrukci klasifikačního stromu místo matice hodnot centrálních momentů byla použila matice V , která vznikne po SVD rozkladu matice A (popsáno v kapitole 7.1.2). Velikost této matice závisí na použitém k ($461 \times k$). Na obrázku 29 je znázorněn klasifikační strom pro matici V o velikosti 461×48 , tzn. že $k=48$.



Obrázek 29: Klasifikační strom pro matice pseudo-dokumentů V po SVD rozkladu, $k=48$.

8.8.1 Závěr experimentu 8

Jak je vidět na obrázku 29, bylo vyhledávání úspěšné. Zejména pak vyhledávání obrázků reálných stop, které se nacházely ve stejných větvích klasifikačního stromu jako odpovídající protějšky z referenční databáze.

9 Závěr

Úkolem této práce bylo nalezení optimální metody pro automatické vyhledávání v obrazových databázích. Byla k dispozici databáze 454 obrázků ideálních otisků podešví bot (obrázky byli pořízeny triviálně, namočením podešve boty do černé barvy, obtištěním na papír a následným nasnímáním a převedením od digitální formy). Pro vyhledávání byly zvoleny tři metody:

1. **Metoda LSI (*Latent Semantic Indexing*)**. Databáze obrázků se předpracovala do předem navržené struktury - indexu a v databázi se následně vyhledávalo za pomoci kosinové míry podobnosti. Tato metoda je ze všech zkoumaných metod časově nejnáročnější. Výsledek byl vypočítán od 5-15 minut.
2. **Metoda DFT (*Discrete Fourier Transform*)**. Spektrální analýza, kde se spočítalo ke každému obrázku z databáze Fourierovo spektrum a potom se tyto spektra porovnávaly pomocí korelačního koeficientu se spektrem vyhledávaného obrázku. Tato metoda vyhledávání je narozdíl od metody LSI velmi rychlá, výsledek je spočítán řádově v sekundách.
3. **Metoda Centrálních momentů**. Zde se spočítaly hodnoty centrálních momentů obrázků databáze a vyhledávaného obrázku. Z těchto hodnot se vytvořila matice centrálních momentů. Pomocí této matice se zkonstruoval klasifikační strom. Tato metoda je též časově nenáročná, výsledek je spočítán řádově v sekundách.

Pro metodu centrálních momentů bylo ke klasifikaci a vyhledávání v databázi využito klasifikačního stromu. Bylo sledováno, jestli se vyhledávaný obrázek nachází ve stejné větvi stromu, jako jeho protějšek z referenční databáze.

Rychlost vyhledávání není jen závislá na použité metodě vyhledávání (resp. množstvím provedených výpočtů), ale i na hardwarovém vybavení počítače, na kterém bylo realizováno. Parametry používaného PC byli:

- CPU: Celeron M 1,5GHz
- RAM: 512+256MB 333MHz
- HDD: Kapacita 40 GB, 5400 rpm, přístupová doba 12 ms, rozhraní Ultra ATA/100, 8-MB cache buffer.

Vyhledávání pro metody LSI A DFT mělo tři stupně:

- a) Vyhledávaný obrázek byl natáčen na intervalu $\pm 10^\circ$ od svislé osy.
- b) Kvalita vyhledávaného obrázku byla degradována subjektivním odmazáním určitých ploch, přidáním Gaussovského bílého šumu a současně byl natáčen na intervalu $\pm 10^\circ$ od svislé osy.
- c) K vyhledávání byl použit reálný otisk (sejmutý přímo z podlahy) stopy

Každá použitá metoda měla své klady i zápory, ať už v úspěšnosti vyhledávání, tak časové náročnosti. Při hledání reálné stopy se však nepodařilo nalézt metodu, která by fungovala se 100% přesností.

Ve vyhledávání z části zůstává subjektivní složka lidského faktoru. Zejména pak při předzpracování (úprava kontrastu, jasu, odmazání přebytečných tmavých ploch) reálného otisku, která je nutná ke zkvalitnění vyhledávání.

Při vyhledávání poškozených stop z referenční databáze dosahovala nejpříznivějších výsledků metoda DFT. Nejlepších výsledků se při vyhledávání reálných stop dosáhlo v experimentu 8, kde bylo využito klasifikačního stromu a matice V , která vznikla po SVD rozkladu rozšířené matice A . V tomto případě byly výsledky vyhledávání obrázků reálných stop velmi příznivé. Obrázky reálných stop se nacházely ve stejných větvích klasifikačního stromu jako odpovídající protějšky z referenční databáze.

Hlavním problémem při vyhledávání obrázků reálných stop je kvalita pořízených obrázků, měřítko a natočení. Tyto parametry obrazu jsou ovlivňovány již při jejich snímání a digitalizaci. Dále pak jsou ovlivňovány při předzpracování obrazu před samotným vyhledáváním a klasifikací.

Při dalším případném zkoumání tohoto problému doporučuji zaměřit se na předzpracování vyhledávaného obrazu. Důraz by měl být kladen na filtry odstraňující nežádoucí šum v obraze a „pokročilé“ morfologické operace jako jsou doplnění děr v obraze a odstranění přebytečných ploch.

10 Literatura

- [1] Mohamed Riyou: *Object Detection using Geometric Invariant Moment*. Science Publications 2006
- [2] Gonzalez R., Woods R., Eddins S.: *Digital Image Processing Using MATLAB*. Pearson Education, Inc. 2004
- [3] Skopal T., Kolovrat M., Snášel V.: *Využití LSI a M-stromu při indexování a vyhledávání obrázků*.
Dostupné na World Wide Web: <http://urtax.ms.mff.cuni.cz/~skopal/pub/znal2005.pdf>
- [4] Rektorys K.: *Přehled užití matematiky I*. Prometheus 2004
- [5] Dostupné na World Wide Web: <http://apfyz.upol.cz/ucebnice/down/mini/fourtrans.pdf>
- [6] Maršálek L., Skapa J.: *Porovnání jednotlivých integrálních transformací ke kompresi obrázků*. 2003 Dostupné na World Wide Web: <http://www.sendme.cz/sklad/index.html>
- [7] Tunák M.: *Využitie texturných charakteristik pre automatickú kontrolu tkaných textilií*. TUL 2004
- [8] Hlaváč V.: *Počítačové vidění vs. digitální zpracování obrazu. Digitální obraz a jeho vlastnosti*. ČVUT Dostupné na World Wide Web: <http://cmp.felk.cvut.cz/~hlavac/Public/TeachingLectures/Uvod+DigObraz.pdf>
- [9] Ličev L.: *Rozpoznání zájmových objektů systému FOTOM*. Acta Montanistica Slovaca Ročník 9 (2004), číslo 2, 96-105
- [10] Houdek P., Schwarz J., Václav Snášel V.: *Intelligentní vyhledávání v dokumentografických informačních systémech*. Dostupné na World Wide Web: <http://www.ikaros.cz/dokumenty/amphor.pdf>
- [11] Skopal T.: *Efektivní vyhledávání ve vektorovém modelu DIS*. VŠB Ostrava, Dostupné na World Wide Web: <http://www.cs.vsb.cz/dis/prispevky/20040325/metricdis.pdf>
- [12] Chazal P., Flynn J., Reilly R.: *Automated Processing of Shoesprint Images Based on the Fourier Transform for Use in Forensic Science*. IEEE Computer Society 2005
- [13] Rui Y., Juany T.S.: *Image Retrieval: Current techniques, Promising Directions, And Open Issues*. Journal of Visual Communication and Image Representation, Vol. 10, No. 1, 39-62, 1999.
- [14] Dostupné na World Wide Web: <http://mdg.vsb.cz/jdolezal/Pgrafika/Prednaska/ZaklBarevneSys.html>
- [15] Dostupné na World Wide Web: <http://komzak.webz.cz/old/skola/dp/sum.html>

- [16] Dostupné na World Wide Web:
http://dsp.vscht.cz/konference_matlab/matlab01/eisenst.pdf
- [17] Dostupné na World Wide Web:
http://radio.feld.cvut.cz/courses/X37KTR/1_Zaklady_radiove_komunikace.pdf
- [18] Dostupné na World Wide Web: <http://sweb.cz/maninf/>

11 PŘÍLOHY

11.1 Příloha 1: *LSI.m* – M-File pro metodu LSI

```
k=32;
name='.tif';
%vytvoření matice dokumentů A
for i=1:454
    I=imread([num2str(i) name]);
    I=double(I);
    [m n]=size(I);
    a=reshape(I, (m*n), 1);
    A(:,i)=a;
end

%SVD rozklad
[U,S,V] = svds(A,k);

%načtení vyhledávaného obrázku
query=imread('23.tif');
figure,imshow(query,[])
query=double(query);

%natočení vyhledávaného obrázku
G=abs(query-255); % negativ obrázku
G2=imrotate(G,-10,'crop'); % pootočení obrázku
G3=abs(G2-255); % pozitiv obrázku
figure,imshow(G3,[])

%přidání Gaussovkého šumu
G4=uint8(G3);
G5=imnoise(G4,'gaussian',0.2,0.2);
figure,imshow(G5,[]);
query=G5;
query=double(query);
query=reshape(query, (m*n), 1);
qc=query'*U*inv(S);

%kosinová míra podobnosti
for j=1:454
    sim(j)=(qc*V(j,:)) / (norm(qc)*norm(V(j,:)));
    pomsim(j)=sim(j);
end
[C,I] = max(sim);
mira_podobnosti=C

%Vykreslení grafu hodnot měř podobnosti
figure
plot (sim(1,1:454),'-','DisplayName','sim(1,1:454)',
'YDataSource','sim(1,1:454)')
text1=['    mira podobnosti = ' num2str(C) '; index = '
num2str(I)];
text(I,C,text1,'HorizontalAlignment','left')
```

11.2 Příloha 2: DFT.m – M-File pro metodu DFT

```
name='.tif';
%DFT pro obrázky databáze
for i=1:461
    I=imread([num2str(i) name]);
    I=double(I);
    I2=I;
    PP=I2;
    [r s]=size(I2);
    mmax=max([r s]);
    mmin=min([r s]);
    IFT=fftshift(fft2(PP));
    S=log(1+abs(IFT));
    [rr ss]=size(S);
poka=S;
for J=1:rr
    for K=1:ss
        pom=(J-(floor(rr+1)/2))^2+(K-(floor(ss+1)/2))^2;
        if pom>(rr/2)^2
            poka(J,K)=0;
        end;
    end;
end;
poka=mat2gray(poka); %%spektrum DFT v úrovních šedi
poka1=im2bw(poka,0.7); %% oprahované spektrum DFT v úrovních šedi

CELL{i,1}=i;
CELL{i,2}=I2;
CELL{i,3}=poka1;
end

CELLv={};
citac=0;
for ii=-10:10 %interval natočení vyhledávaného obrázku
    citac=1+citac;
    stupennat(citac)=ii;
    Ivyhl=imread('114.tif');

    %natočení vyhledávaného obrázku
    Ivyhl=abs(255-Ivyhl);
    Ivyhl=imrotate(Ivyhl,ii,'crop');
    Ivyhl=abs(255-Ivyhl);

    %přidání Gaussovkého šumu
    Ivyhl=imnoise(Ivyhl,'gaussian',0.2,0.2);
    Ivyhl=double(Ivyhl);

    Iv=Ivyhl;
    [r s]=size(Iv);
    mmax=max([r s]);
```



```

mmin=min([r s]);

%DFT pro vyhledávaný obrázek
IFT=fftshift(fft2(Iv));
S=log(1+abs(IFT));
[rr ss]=size(S);
poka=S;
for J=1:rr
    for K=1:ss
        pom=(J-(floor(rr+1)/2))^2+(K-(floor(ss+1)/2))^2;
        if pom>(rr/2)^2
            poka(J,K)=0;
        end;
    end;
end;
poka=mat2gray(poka);
poka1=im2bw(poka,0.7);
CELLv{1,citac}=ii;
CELLv{2,citac}=Ivyhl;
CELLv{3,citac}=poka1;
end

CMAT=[];
% korelace obrázků spekter DFT
for iii=1:454
    for jjj=1:citac
        CMAT(iii,jjj)=corr2(CELL{iii,3},CELLv{3,jjj});
    end
end

for jjj=1:citac
    [rr bb]=max(CMAT(:,jjj));
    sloupmax1(jjj,1)=stupennat(1,jjj);
    sloupmax1(jjj,2)=rr;
    sloupmax1(jjj,3)=bb;
end
MM=max(CMAT(:));
[m n]=find(CMAT==MM);

%seřazená maxima
CMAT1=[];
for kk=1:454
    CMAT1(kk,:)=sort(CMAT(kk,:), 'descend');
end
[B,IX] = sort(CMAT1(:,1), 'descend');
qq=[B IX];

```

11.3 Příloha 3: CMtree.m – M-File pro klasifikační strom z matice centrálních momentů databáze

```
name1='.tif';
pole=[];

%vytvoření matice centrálních momentů rozšířené databáze
for i=1:461
    I=imread([num2str(i) name1]);
    phi=abs(log(invmoments(I)));
    pole=[pole; phi i];
end
index=num2str(pole(:,end));

%Klasifikační strom z matice centrálních momentů rozšířené
databáze
T = treefit(pole(:,1:end-1),index)
treedisp(T)
```

11.4 Příloha 4: LSItree.m – M-File pro klasifikační strom z matice V $k=48$;

```
name='.tif';
%vytvoření matice dokumentů A
for i=1:461
    I=imread([num2str(i) name]);
    I=double(I);
    [m n]=size(I);
    a=reshape(I, (m*n), 1);
    A(:,i)=a;
end

%SVD rozklad
[U,S,V] = svds(A,k);

for i=1:461
    pom(i,1)=i;
end
index=num2str(pom(:,1));

%Klasifikační strom z matice V
T = treefit(V,index)
treedisp(T)
```